****

****

**信息学院软件工程系**

**《JAVA程序设计》实验报告**

大作业-记事本

**姓名：陈澄**

**学号：32420212202930**

**学院：信息学院**

**专业：软件工程专业**

**完成时间：2023.06.02**

**一、实验目的及要求**

**二、实验题目及实现过程**

题目1：记事本

1. 实验环境（集成开发环境、jdk版本、字符编码等）

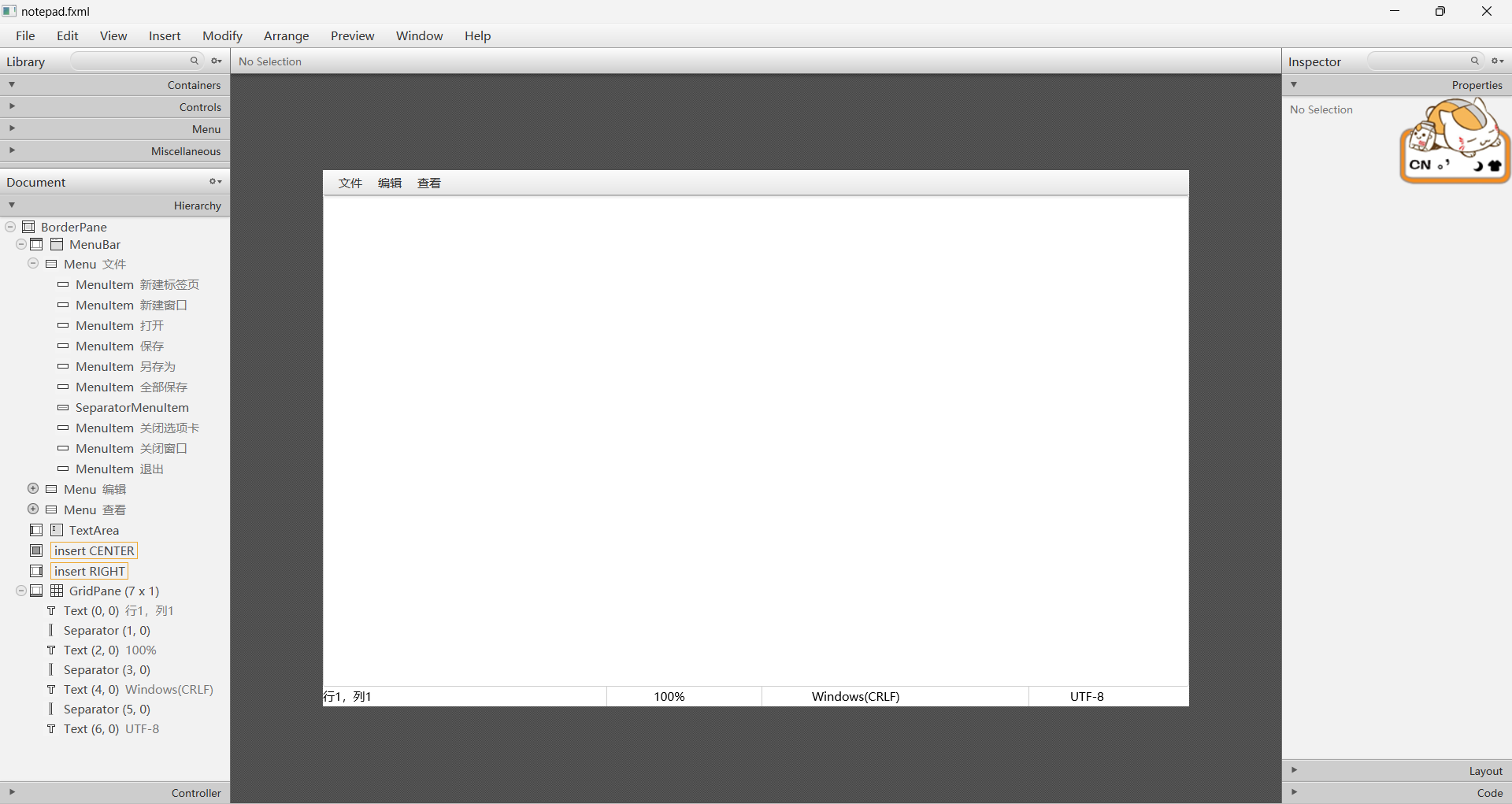
集成开发环境：Intellij IDEA

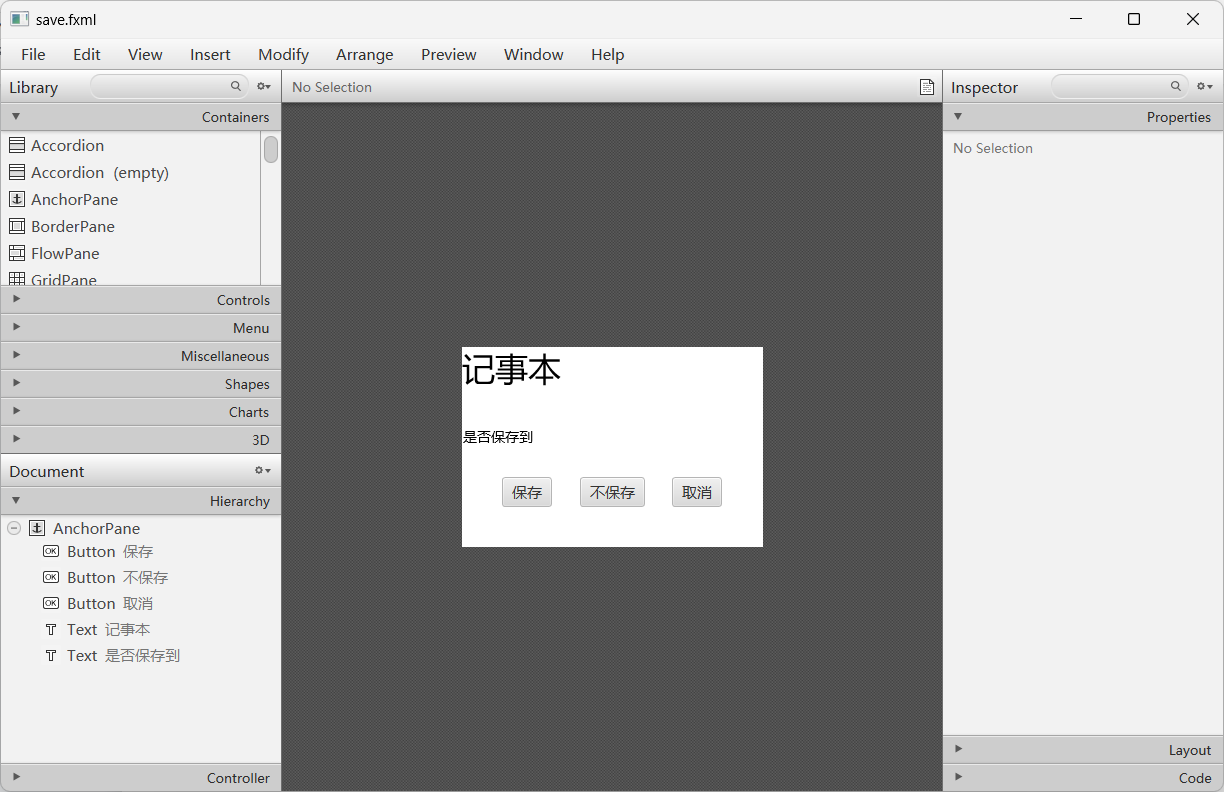
Jdk版本：17.0.5

字符编码：ASCII,UTF-8

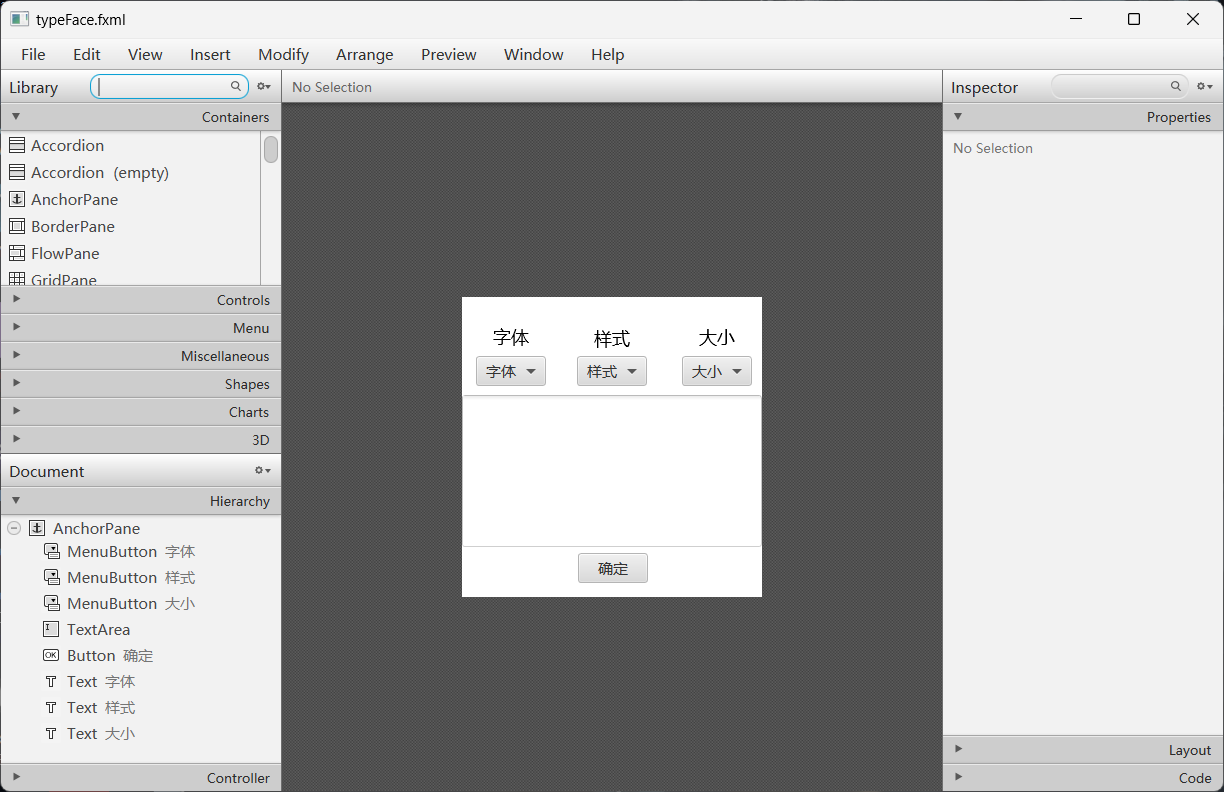
1. 实现过程（本部分为主要评分依据，请描述解题思路，比如总共设计几个类，各个类的用途、成员、主要方法等及其之间调用关系等）
2. 整体思路：将记事本分为四个界面：记事本主页面，保存界面，字体选择界面和查找与替换界面。每个页面都有自己对应的fxmlController脚本控制。各个界面的展示如下：

记事本主页面：

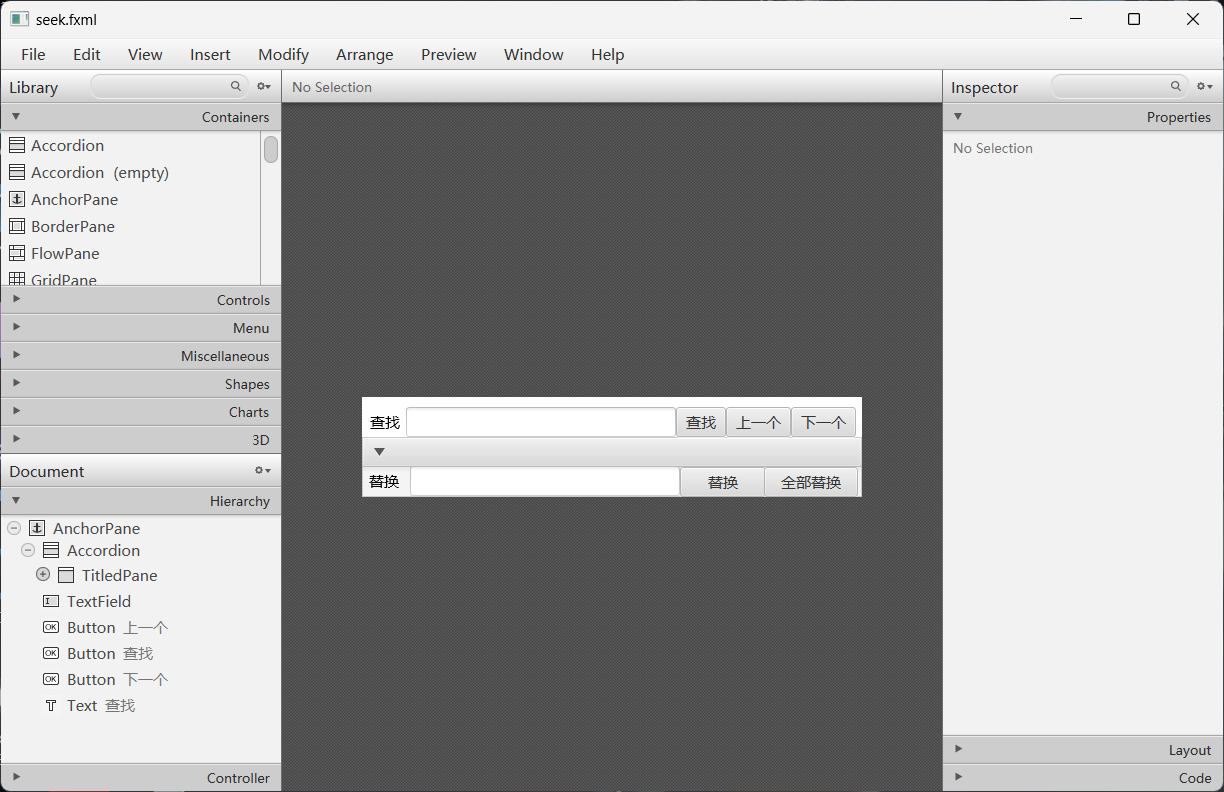


保存界面：

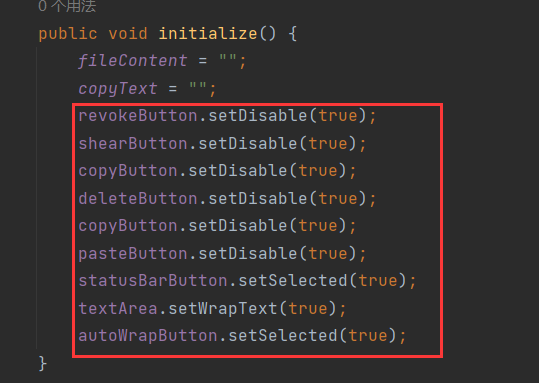
字体选择界面：



查找与替换界面：



1. 下面对一些关键功能的脚本进行展示与分析：
2. 按钮灰度：初始化的代码中通过menuitem下的setDisable方法先将相关按钮全部设置为不可用。



此外，将按钮灰度的重新判断放置在相关menu的OnShowing方法中，也就是当菜单展开时会调用该方法进行判断灰度。该部分通过TextArea里面的getSelectText并判断返回的串是否为空来判断是否有文本被选中，并据此改变大部分按钮灰度。撤回按钮通过TextArea下的isUndoable判断是否可以撤回。粘贴按钮通过判断copyText（也就是被复制字符串保存的临时变量）为空来判断粘贴按钮的灰度。



1. 文件的保存与读取

读取：通过FileChooser里面的showOpenDialog方法打开一个用于打开文件的窗口。

并获得文件路径并存储在filePath中（后续用于判断是否可以直接保存，或者让用户选择保存位置），将文本内容储存在fileContent中（后续用于判断是否可以直接退出，或者让用户选择是否保存）。读取时将File传入FileReader，再将FileReader传入BufferReader，然后调用BufferReader下的readLine进行逐行读取，并写入字符串s，最后通过TextArea里面的setText方法将s传入。



保存：使用FileChooser里面的ExtensionFilter固定将文件保存为.txt形式，通过其showSaveDialog打开保存窗口，其余部分于文件读取相似。

上述代码为另存为的代码，保存的代码会增加一个对filePath是否为空的判断。



1. 退出：退出功能被点击后，会对fileContent的内容进行判断，如果当前编辑的内容等于上次进行保存的内容，则直接退出，没有保存的必要，如果为否，则会打开保存界面，有三个按钮“保存”，“不保存”，“取消”进行选择。



选择“保存”时直接调用之前文件保存按钮的方法，选择“不保存”则直接退出，选择“取消”则关闭当前页面。此处通过saveText.getScene().getWindow()获取当前窗口saveText是所需要关闭窗口内的随意一个控件，获得其Scene即可再获得其Windows。



1. 撤销剪切复制粘贴删除全选：

撤销：调用TextArea里面的undo方法。

剪切：调用TextArea里面的getSelectedText方法将剪切的文本保存起来，再调用replaceSelection将所选中的内容全部替换为空字符串实现删除的效果。

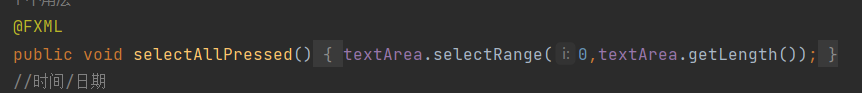
复制：调用TextArea里面的getSelectedText方法将剪切的文本保存起来。

粘贴：将上述获得的复制内容copyText作为参数传入replaceSelection即可（这个方法如果没有选中的内容会在光标后面将传入的内容写入）。

删除：调用replaceSelection将所选中的内容全部替换为空字符串实现删除的效果。

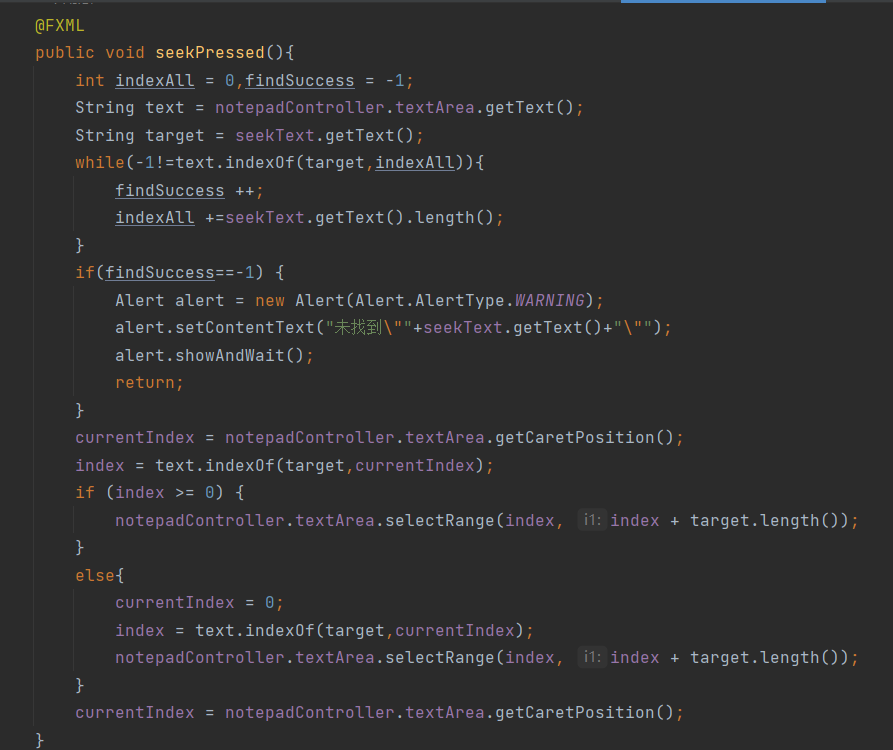
全选：调用TextArea里面的selectRange并传入0,textArea.getLength()将全部内容选中。



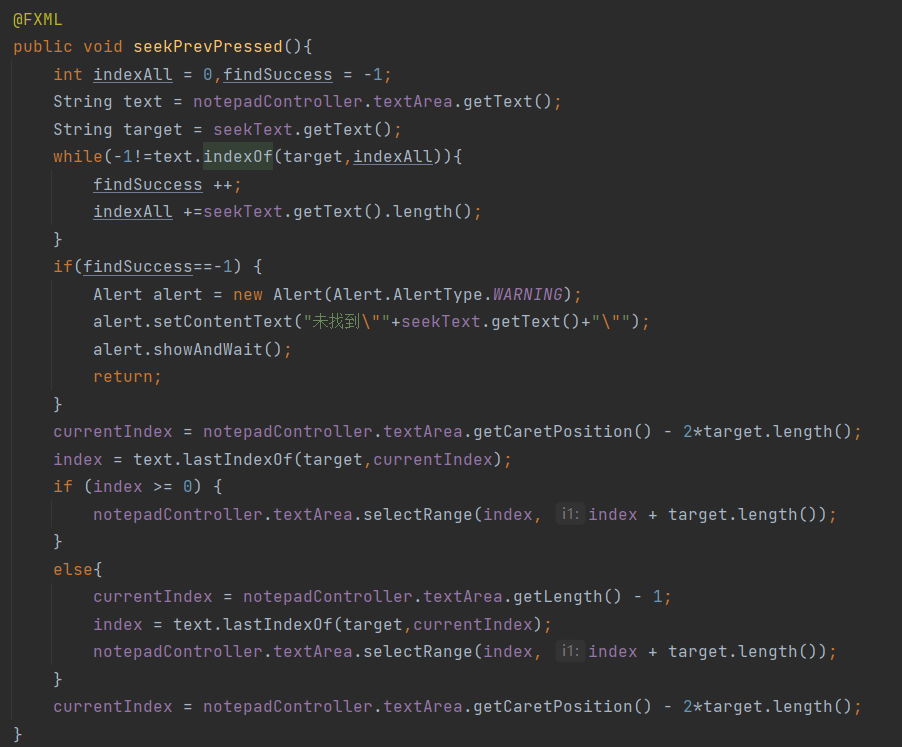


1. 查找与替换：

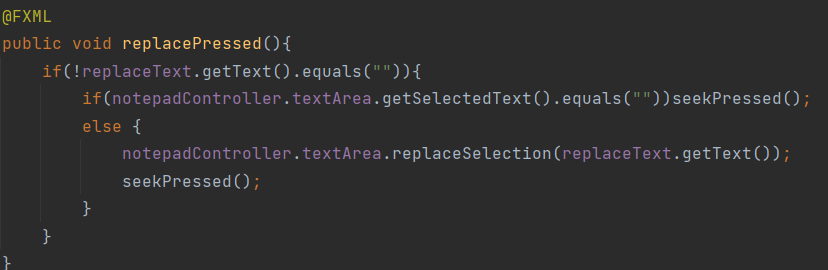
查找：text字符为当前TextArea的内容，target为查找输入框输入的内容。先通过一个while循环总体判断是否可以查找到，若无法查找到则通过Alert弹出警告对话框。其核心代码部分调用text里面的indexOf方法，其传入的第一个字符串是被查找字符串，第二个参数是开始查找的currentindex，每次查找之后将currentindex加上target的长度，实现查找下一个的功能（实际上查找与查找下一个的功能完全一样），如果indexOf的返回值小于0则表示未查找到，此时表示已经查找到最后一个，将currentindex置0回到第一个实现循环查找。



查找上一个：与上述代码相似，将indexOf改为lastIndexOf实现向前查找的功能，此外currentIndex每次查找后减去查找目标的长度，查找到第一个将currentIndex置为textArea.getLength() - 1即最后一个字符，实现循环查找。



替换：调用replaceSelection进行替换再调用查找的方法使光标到达下一个即可。



全部替换：与上述过程类似，外加一个while循环即可。



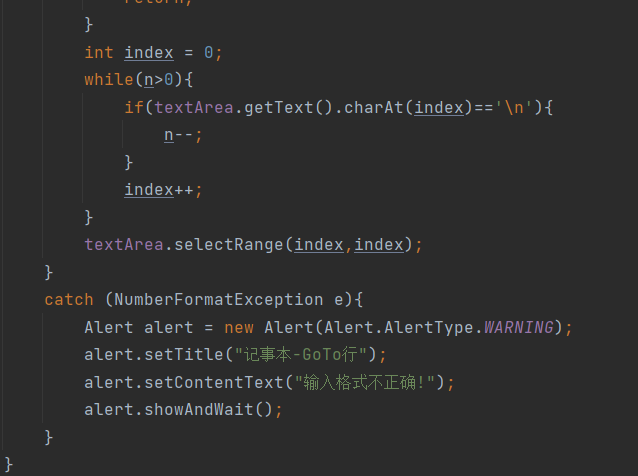
1. 行列判断与转到某行：

行列判断：TextArea里面只有获得光标位置的方法，并没有直接获得具体行列的方法，所以使用判断’\n’字符的个数判断行数，每次获得一个\n将列数置零。该方法在TextArea被鼠标点击获得获得键盘输入时调用（这两种行为会改变行列数）。



转到行数：大致思路与行列判断相似，仍然通过selectRange移动光标到行最前面，传入两个参数相同就不会选中字符而是移动光标。

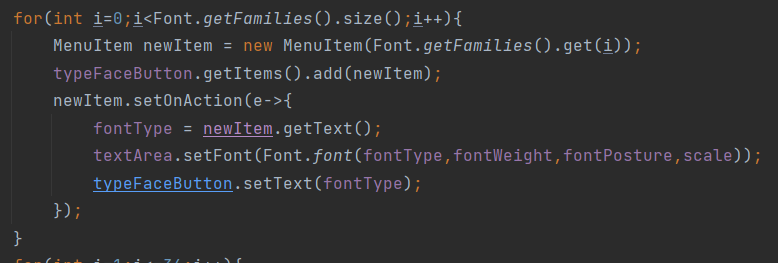




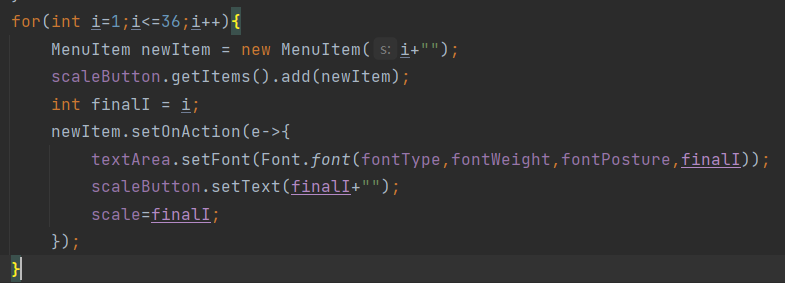
1. 字体与样式

这块的代码直接使用 lambda 表达式设置按钮的点击事件（因为字体实在太多了）

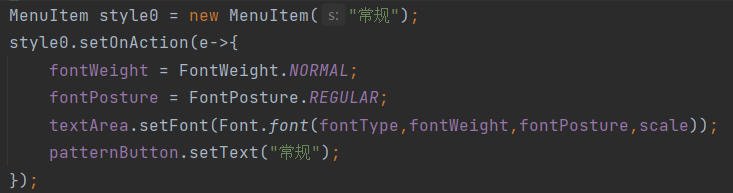
字体：循环生成新的MenuItem并设置其点击事件，通过TextArea内的setFont设置其字体。



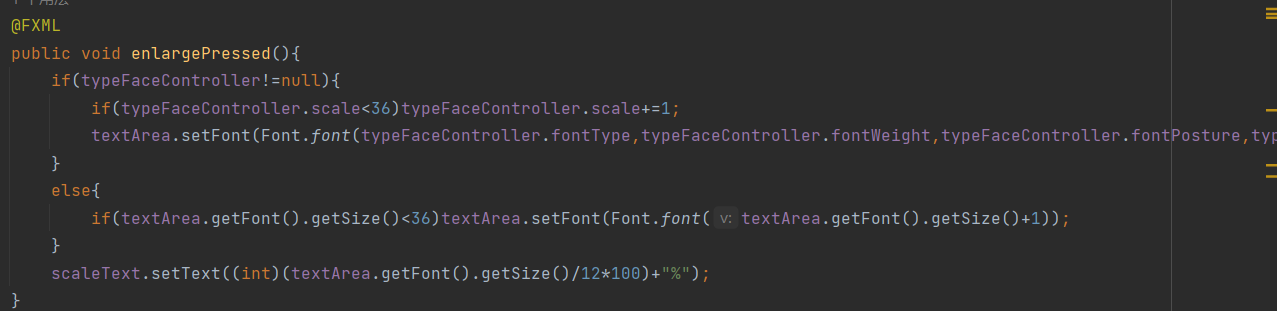
大小：与字体的设置类似，仅设置了1-36的大小（可以拓展）。

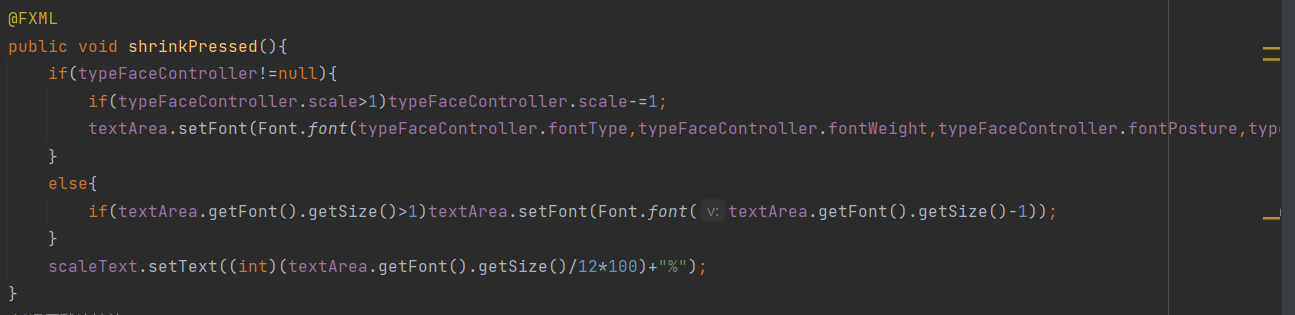


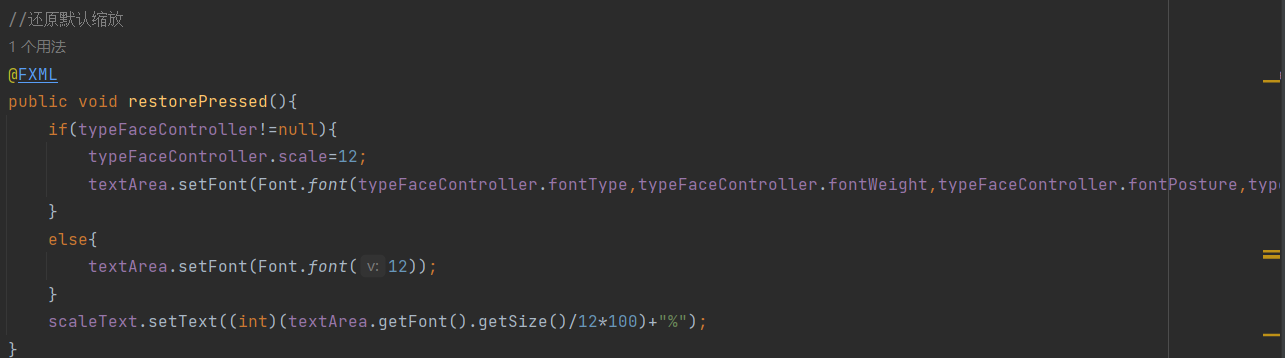
样式：这边就需要一个按钮一个按钮去设置。FontWeight为字体的粗细，有NORMAL,BOLD(粗体),BOLDER(极粗),LIGHT(细体)。FontPosture为字体的板式，有REGULAR和ITALIC(斜体)。



1. 放大缩小还原：该部分通过设置字体大小实现放大缩小的效果，注意最后需要将改动传入状态栏（通过字体大小与12（默认大小）的比例计算）。

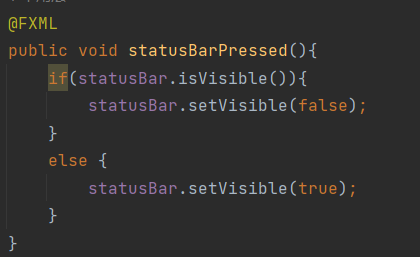






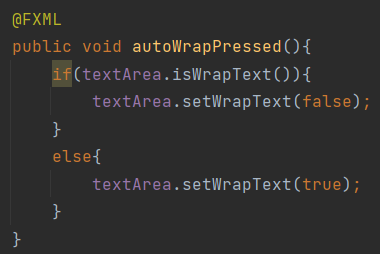
1. 状态栏可视化：

设置其isVisible方法即可



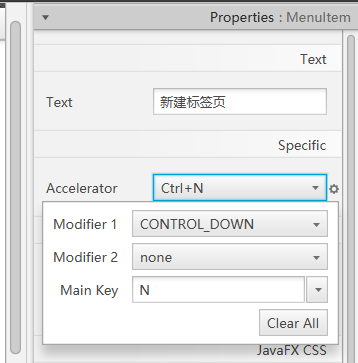
1. 自动换行：

设置TextArea里面的setWrapText即可。

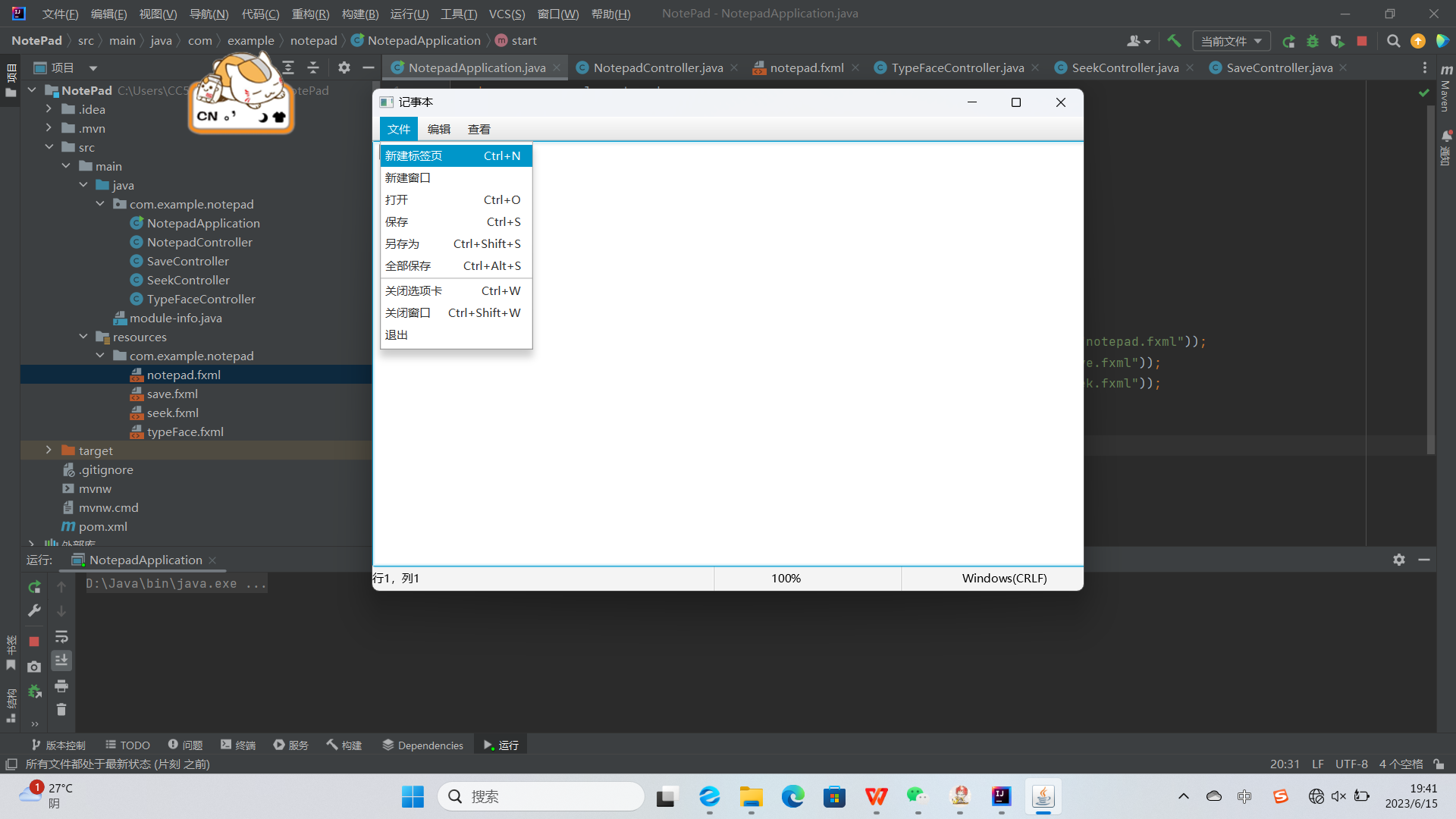


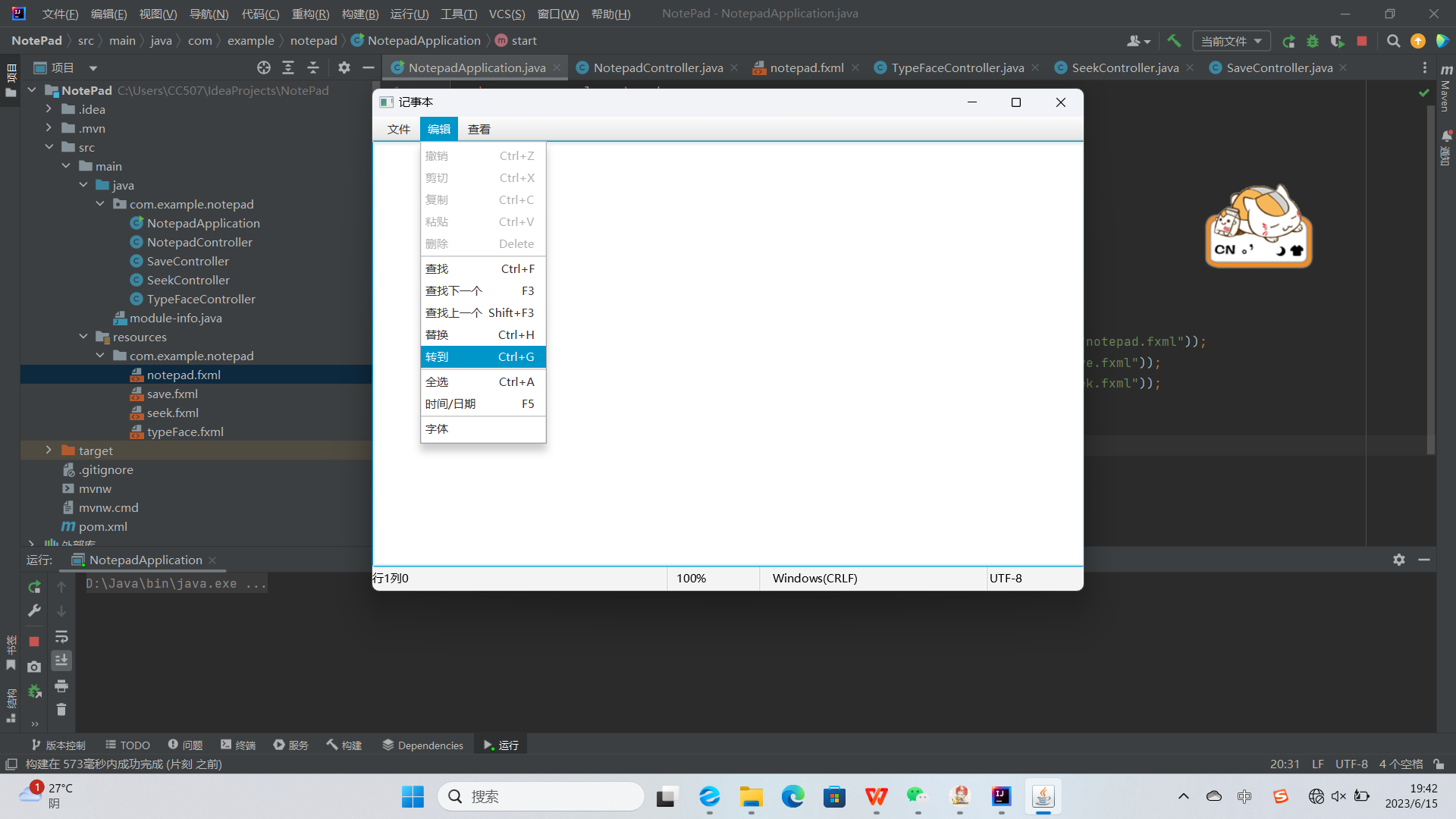
1. 快捷键设置：

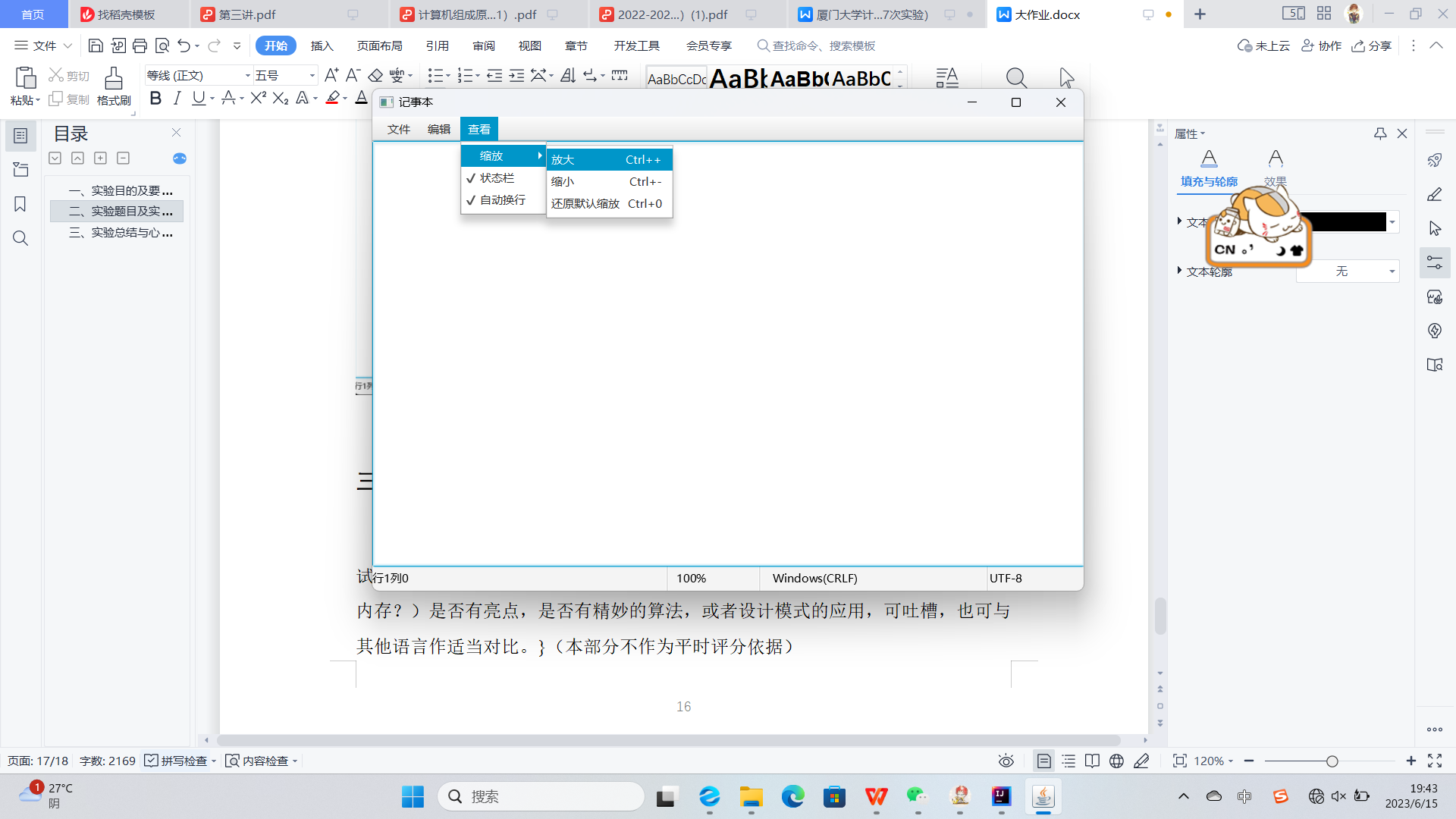
Scene Builder里面的Accelerator里面设置



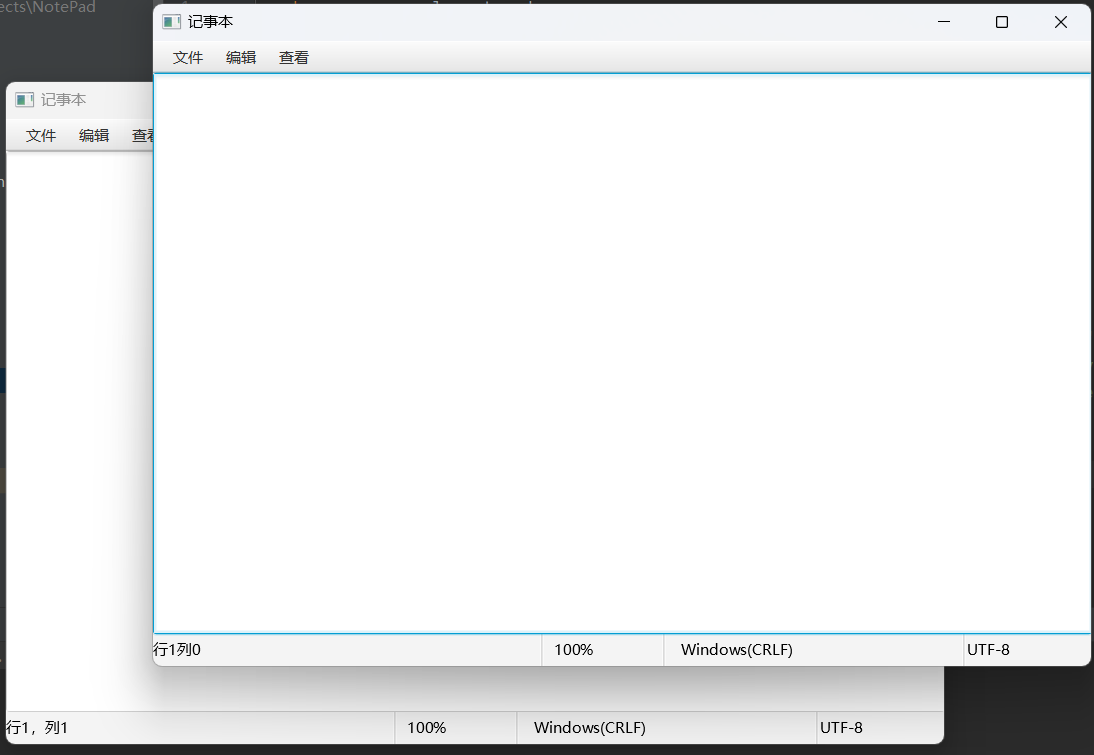
1. 过程截图（本部分为主要评分依据，一张全屏截图（必须）、若干运行结果展示图（可选），主要代码（可选））
2. 菜单以及主页面展示：



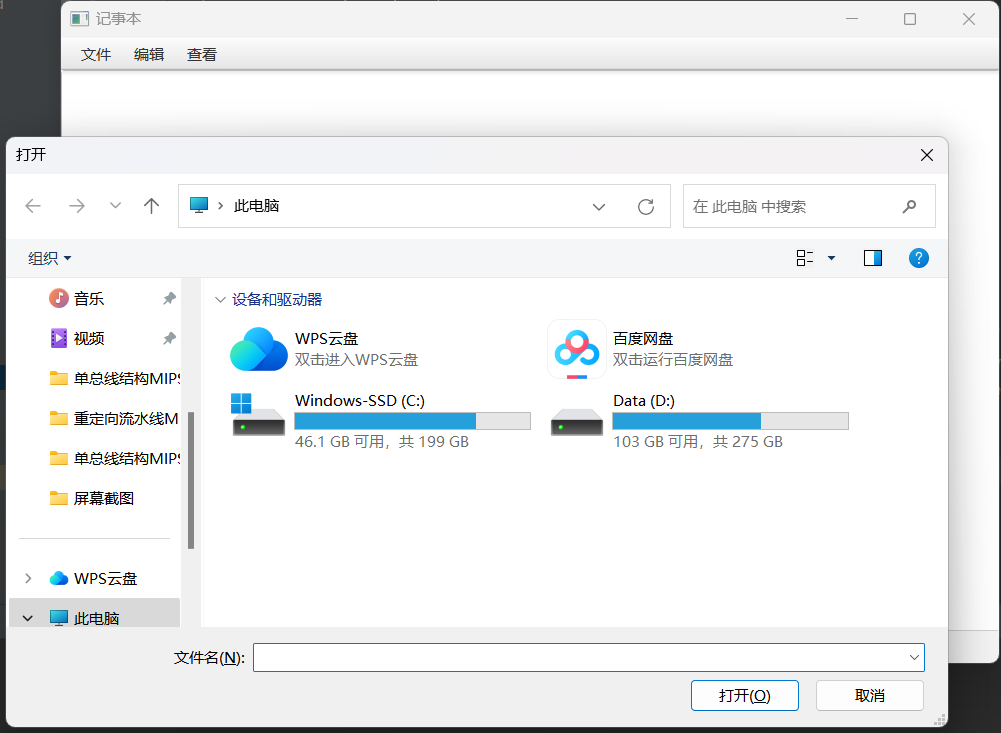


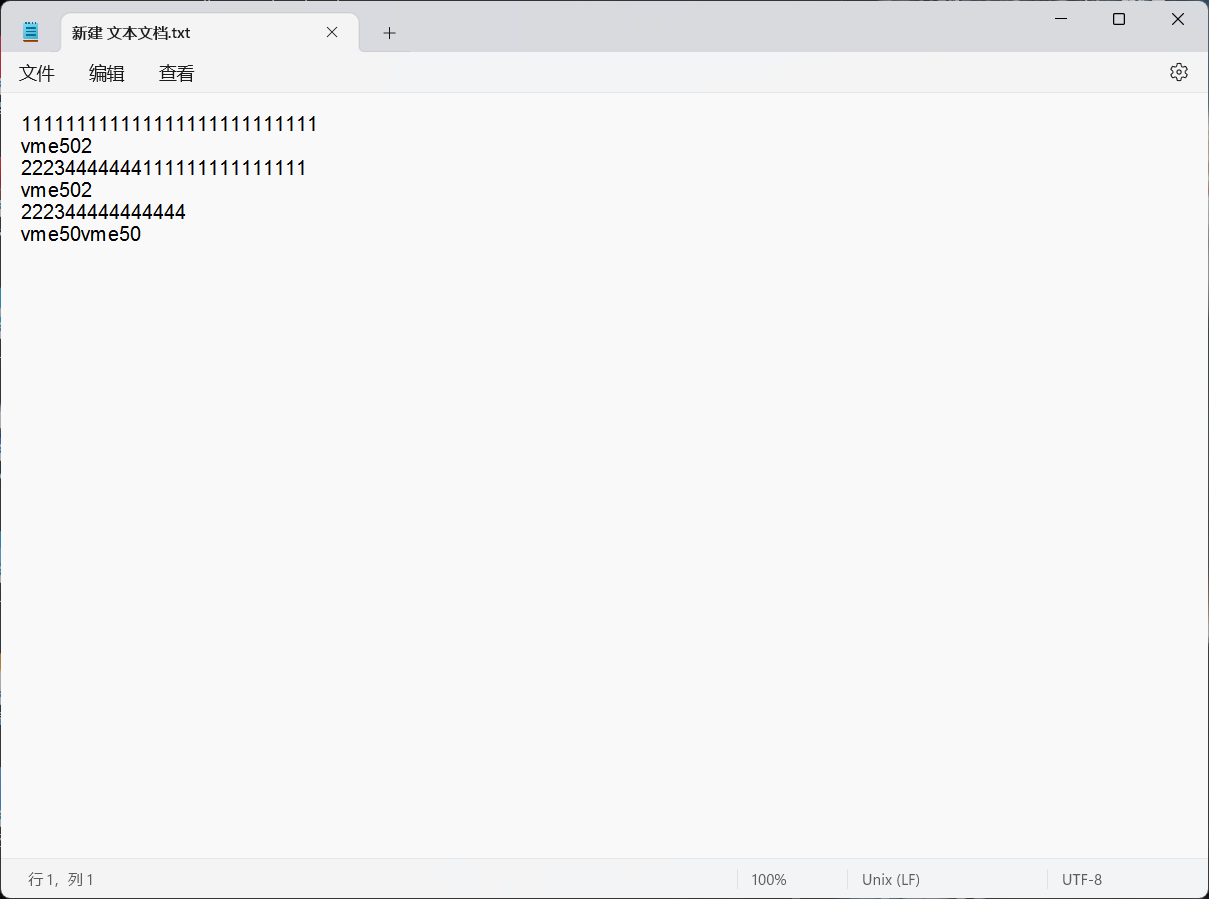
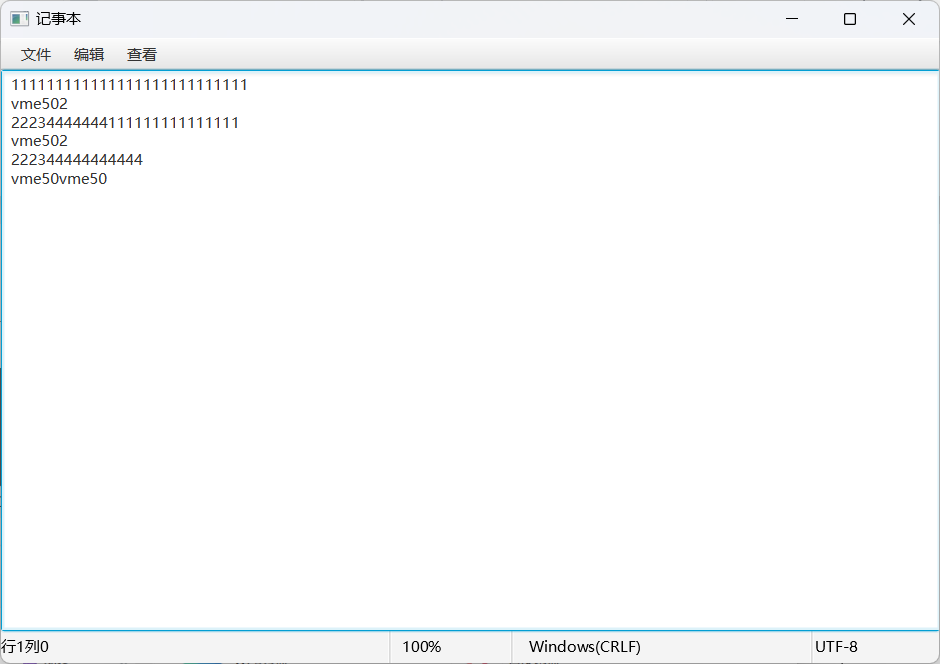


1. 新建窗口：



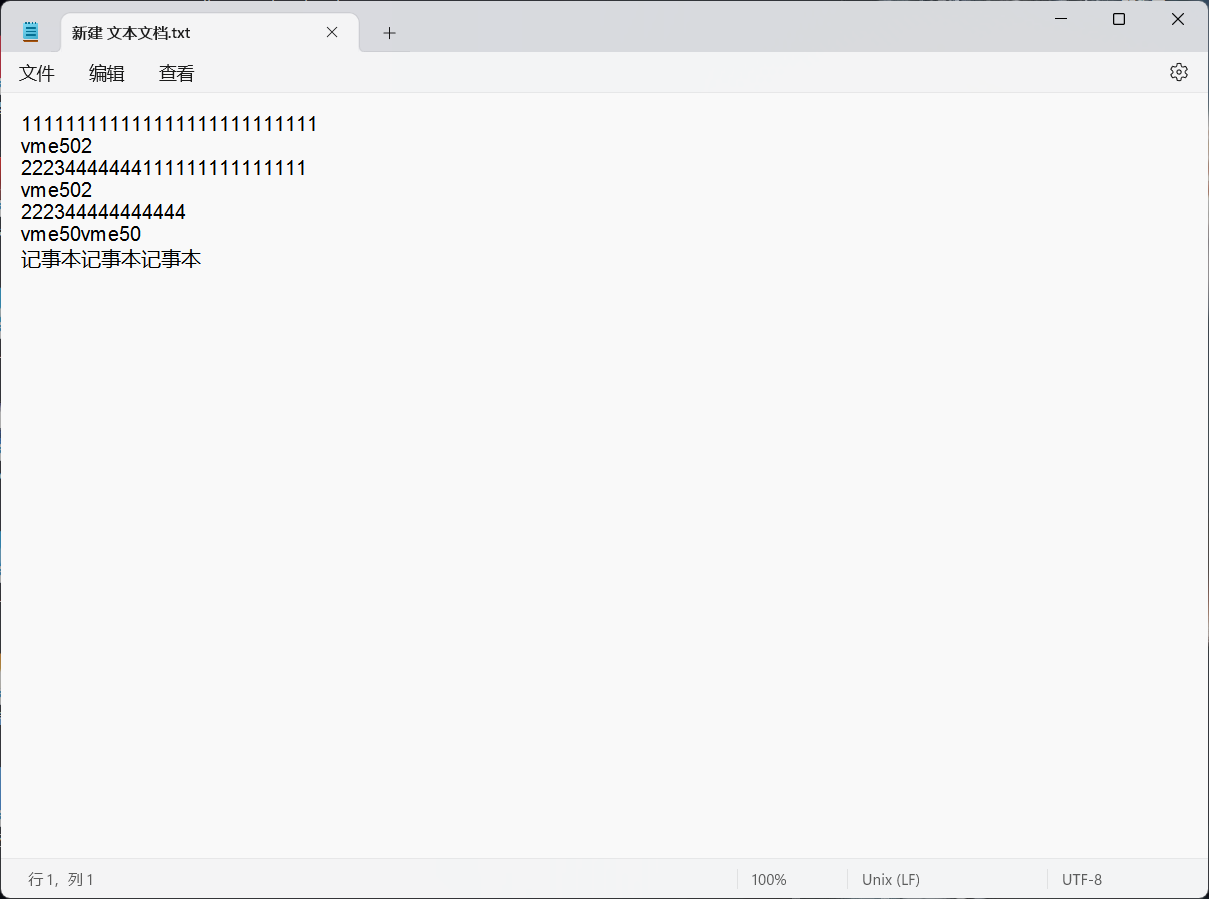
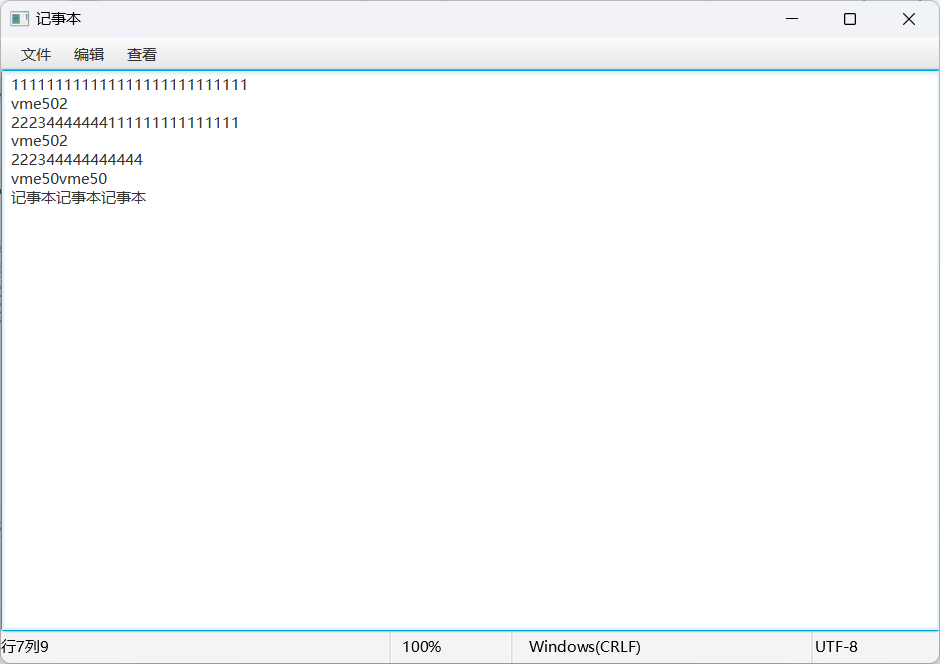
1. 打开文件



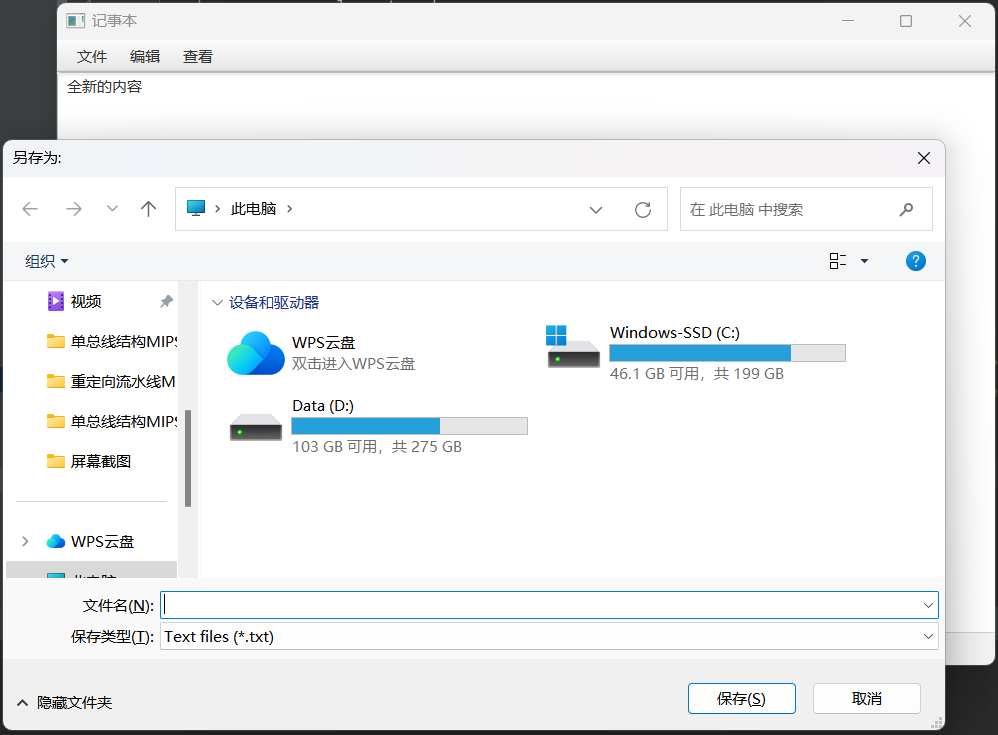


（右侧是真记事本打开的）

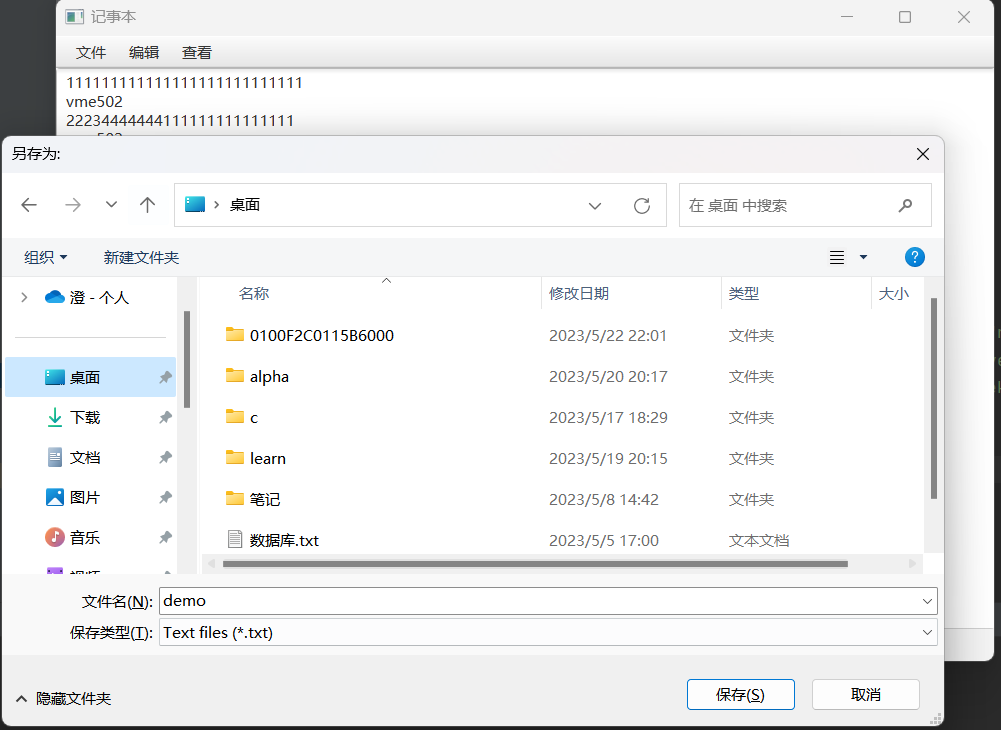
1. 保存：写入几个字符后点击保存，再用真记事本打开（右侧），内容确实正确保存。



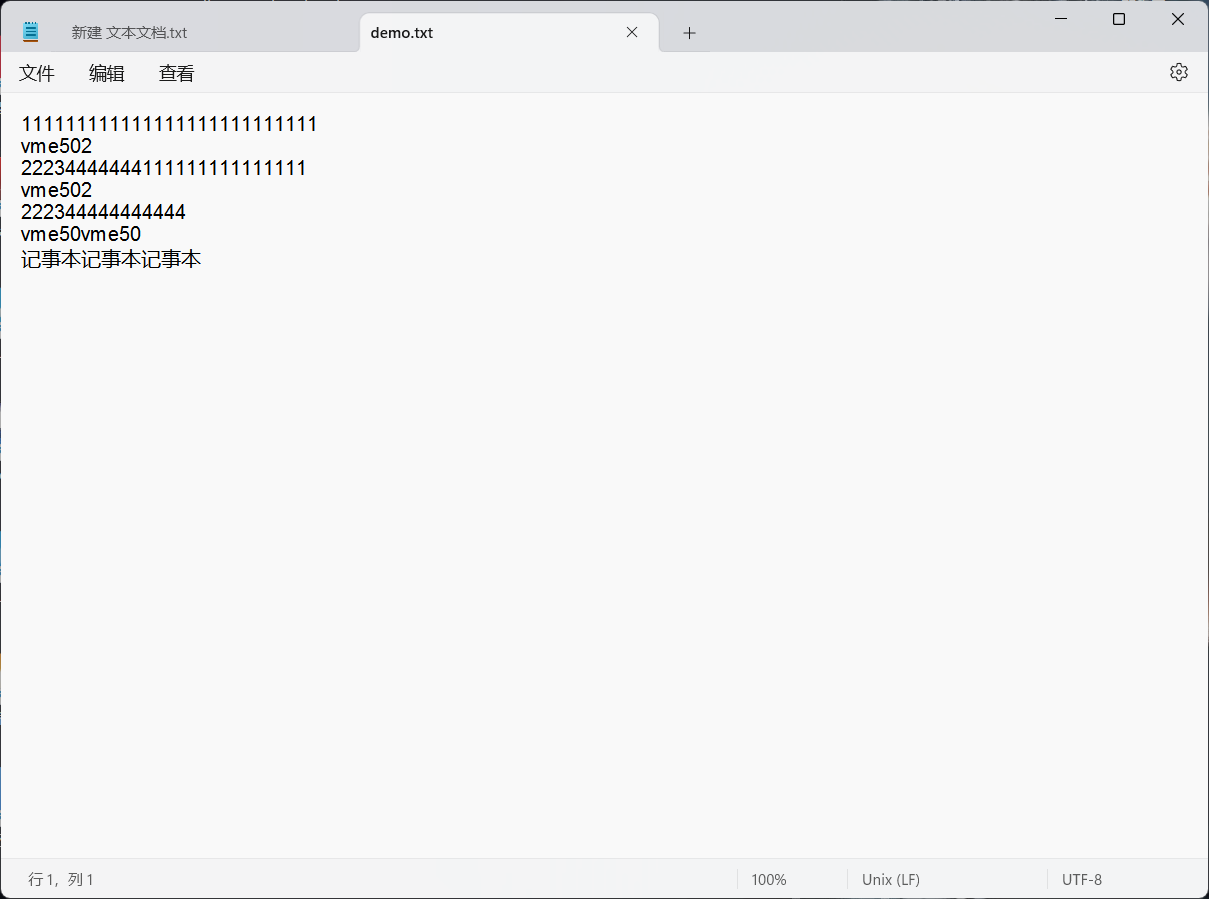
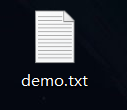
如果不是打开已有的文件而是直接写入内容后点击保存，则会打开另存为窗口：



1. 另存为：点击后打开一个文件选择窗口，可以输入文件名，并且选择路径，但是文件只能保存为.txt格式。



此处我选择在桌面保存，命名为demo，点击保存后桌面正确出现文件，真记事本打开如下：



1. 退出：

如果退出时内容与上次保存一致，则直接退出（就不展示了）

如果不是打开的文件，则打开保存窗口进行询问：



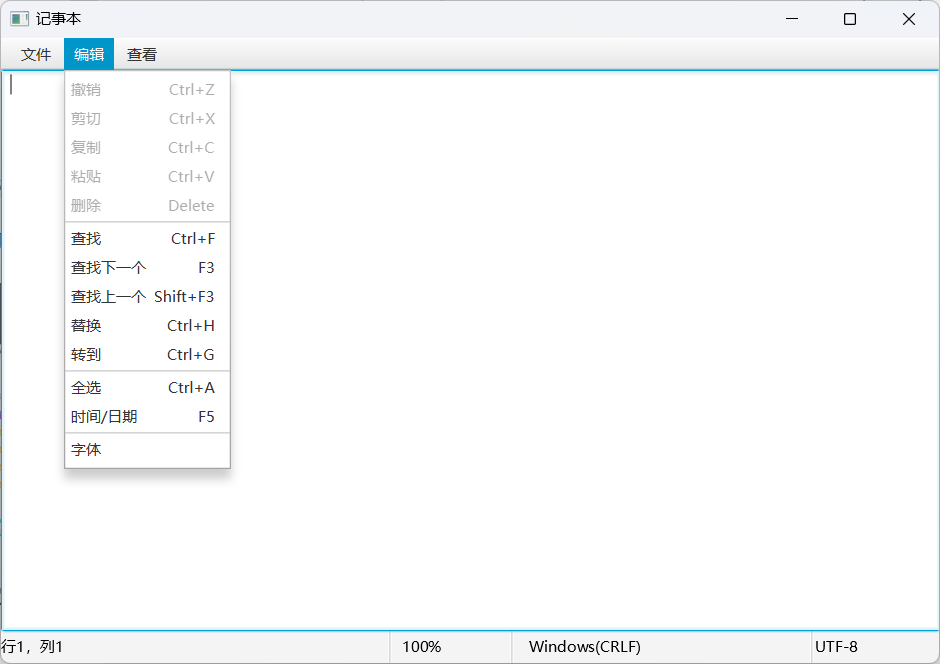
如果是打开的文件，但是内容发生了更改，则询问时会显示上次保存的路径：



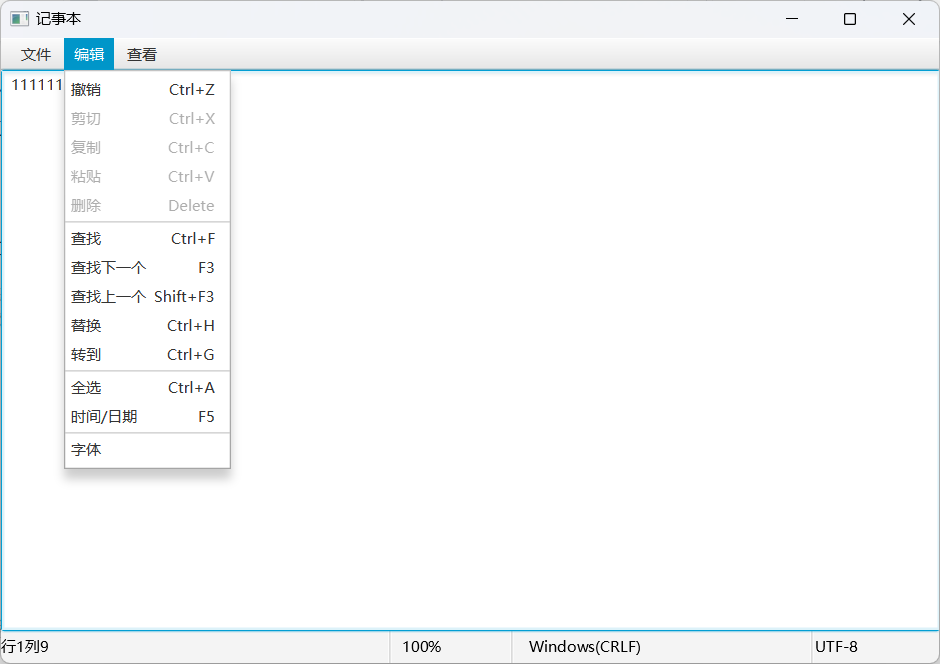
点击×或者取消都会回到记事本界面，点击保存会打开保存界面，如果选择了保存位置则会保存并退出记事本，点击不保存即不保存并且退出记事本。

1. 撤销剪切复制粘贴删除全选：

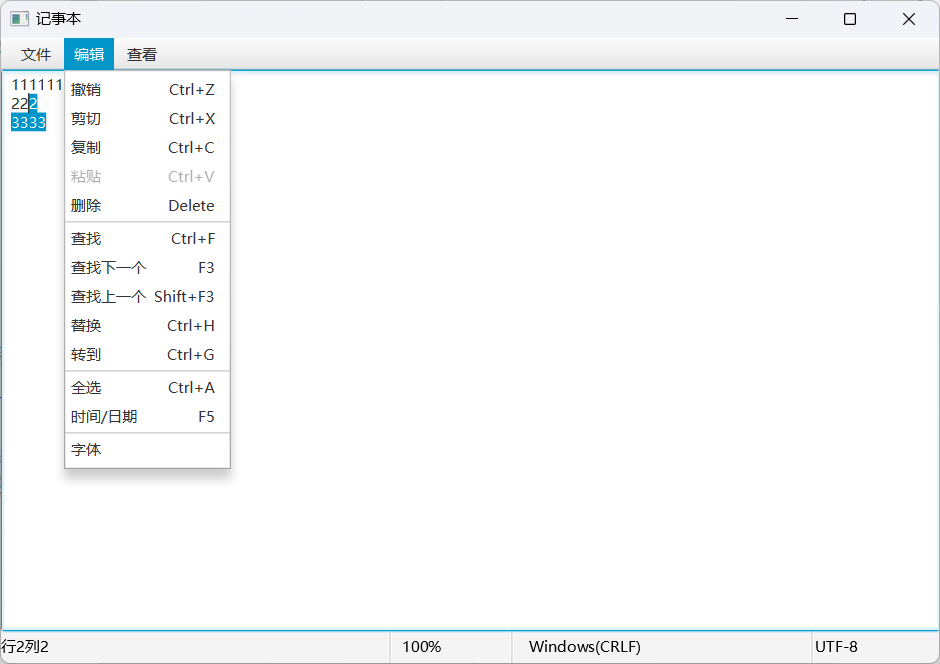
没有任何内容时按钮全灰



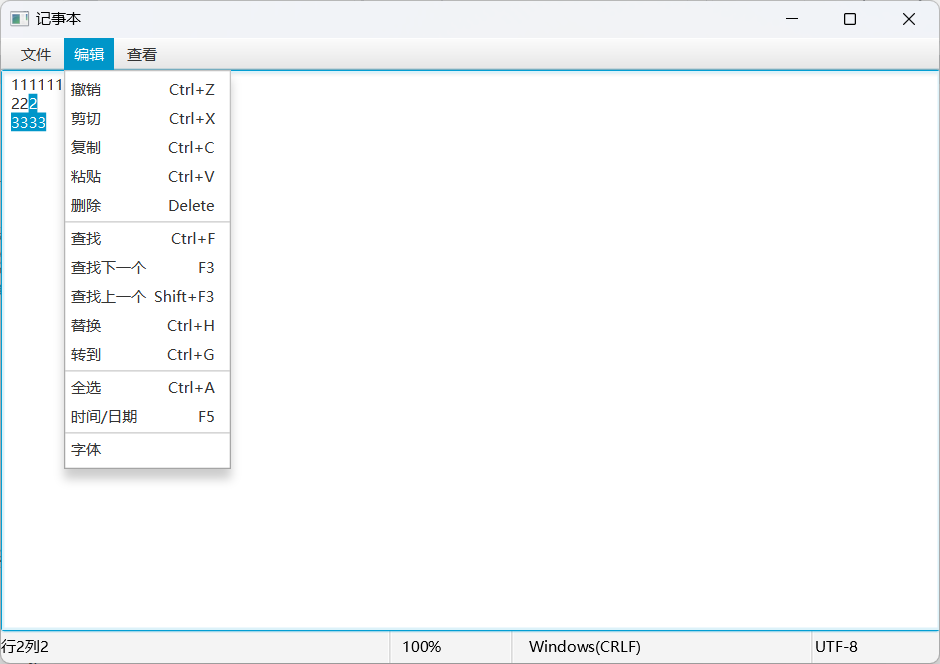
写了内容后可以撤销，其他按钮仍然是灰的



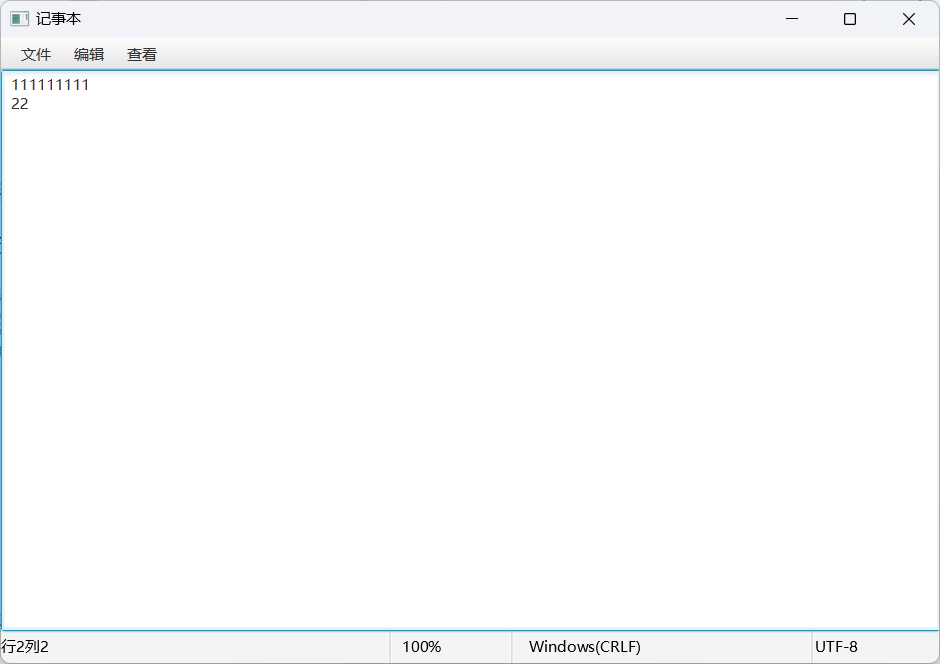
选中了内容后，剪切复制与删除都是亮的，但是粘贴仍是灰的（因为粘贴板没有内容）



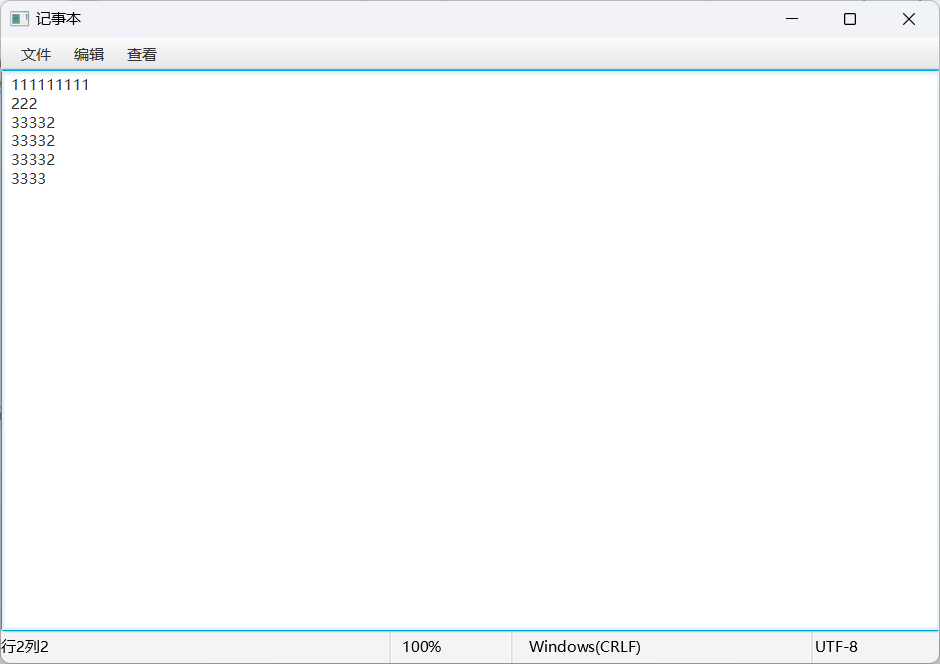
按下复制之后就可以使用粘贴了



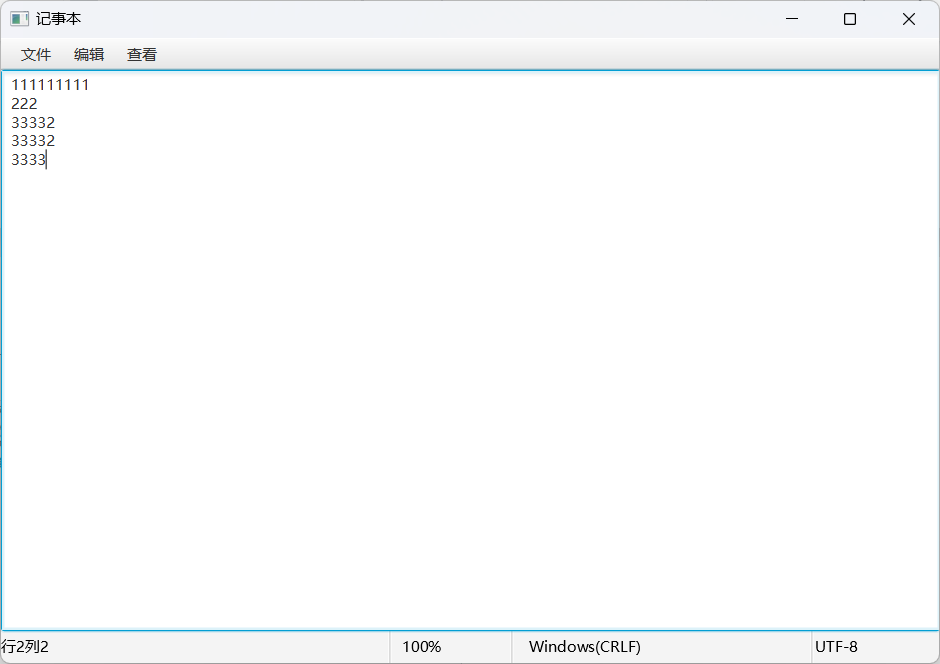
按下剪切（这边效果和删除是一样的，只不过剪切的内容后续可以贴出来）



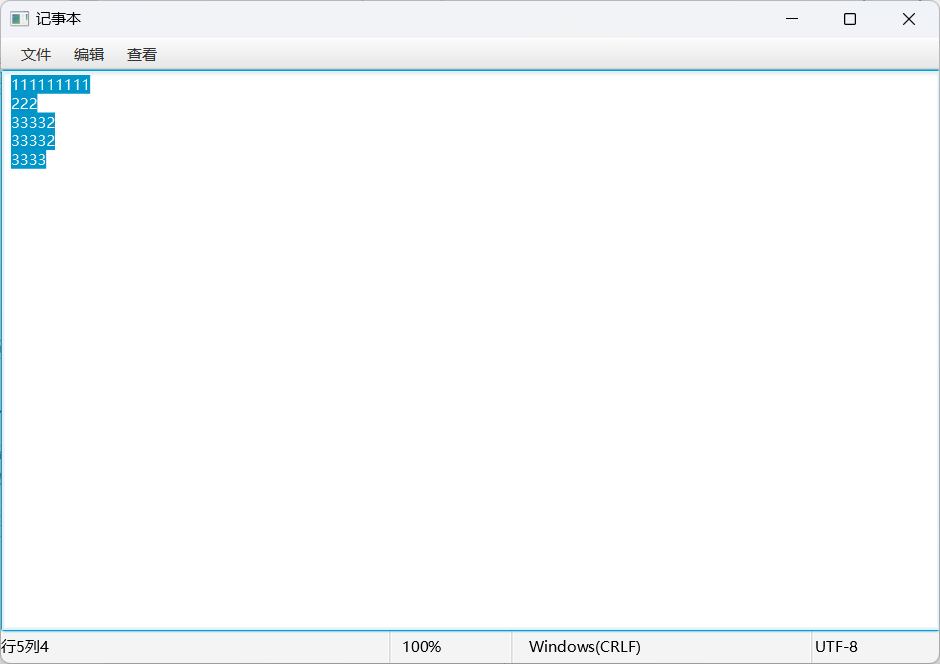
多次粘贴：



点击撤销撤回去了一次粘贴

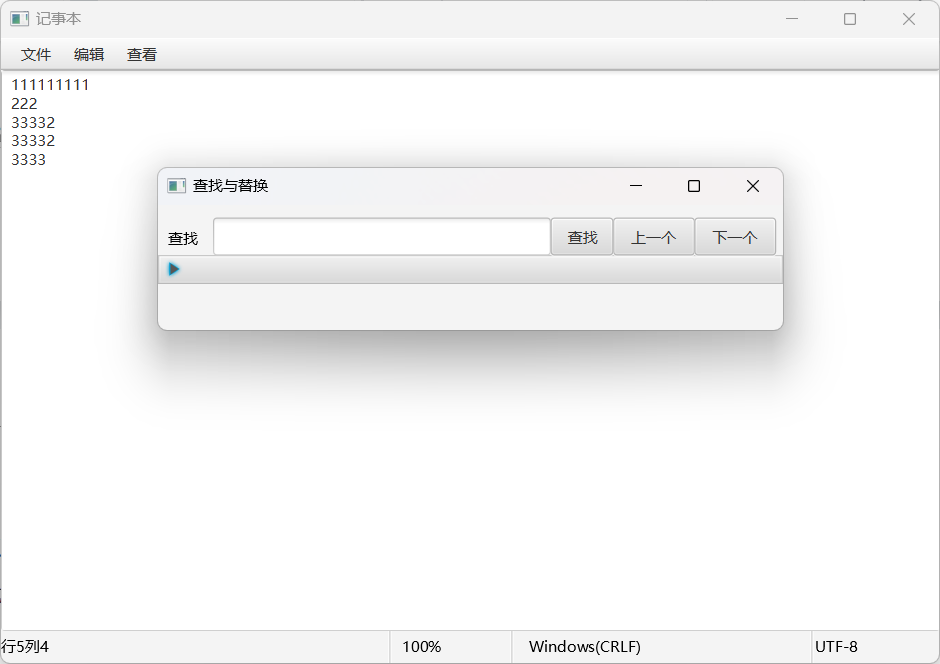


点击全选：

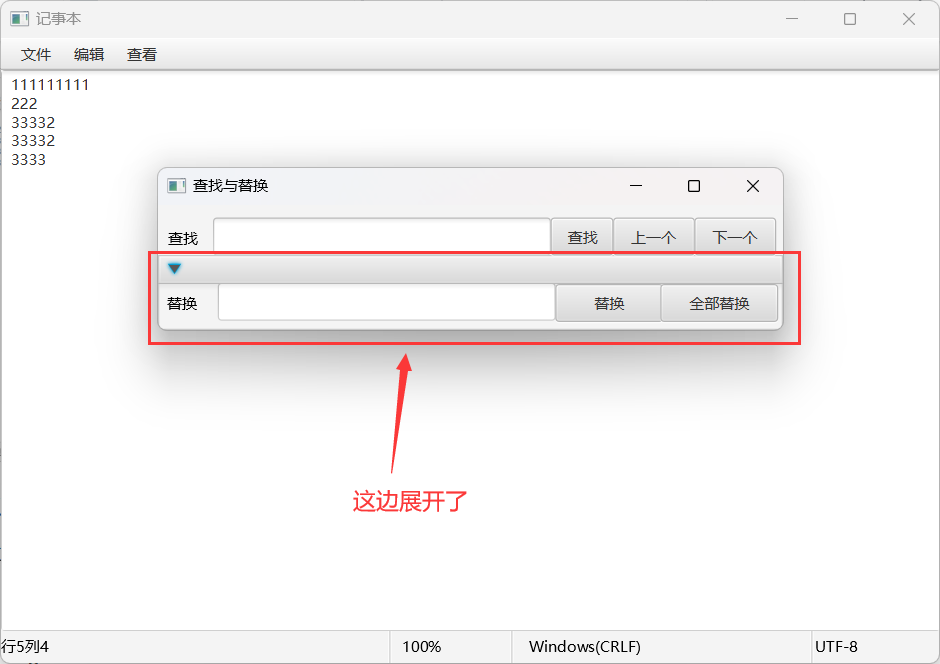


1. 查找与替换

点击查找/查找下一个/查找上一个都会打开查找窗口（如果没打开），如果打开了就是执行对应功能。

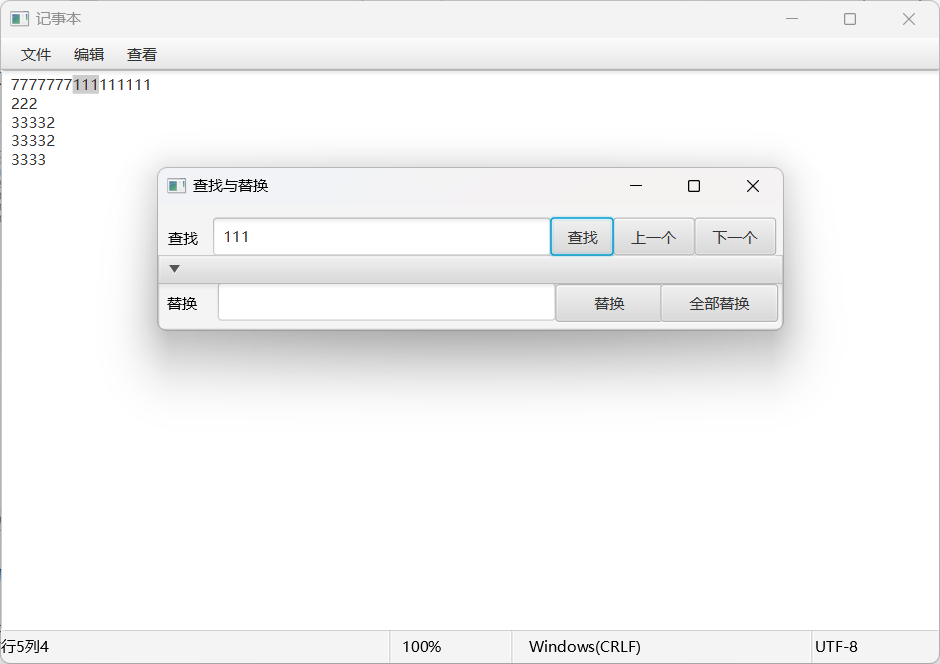


点击替换/全部替换都会打开同样窗口（如果没打开），不同的是他会把替换的折叠栏一起打开，如果打开了窗口再点击就是执行对应功能了。

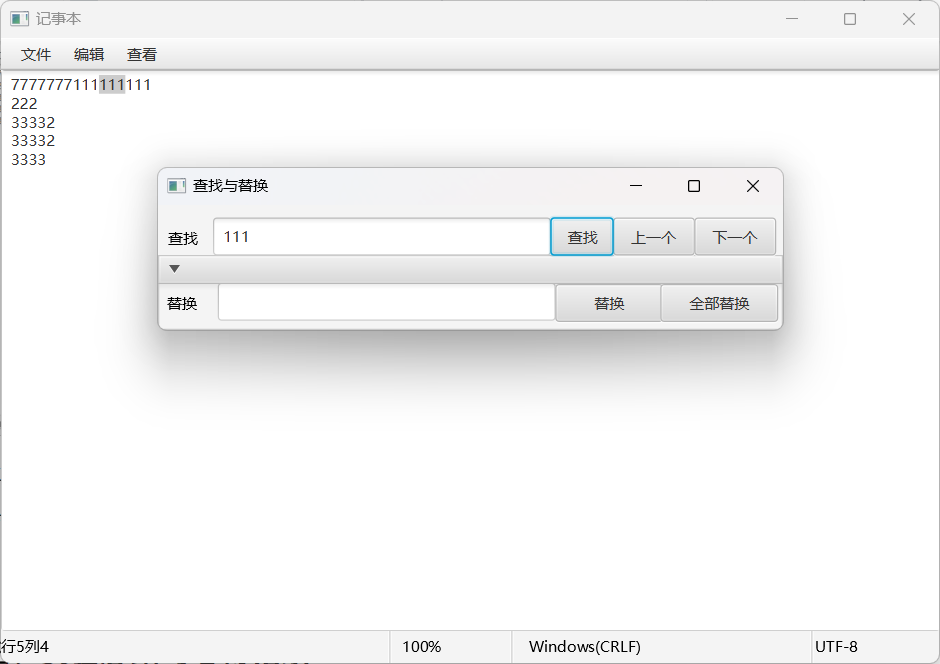


功能演示：

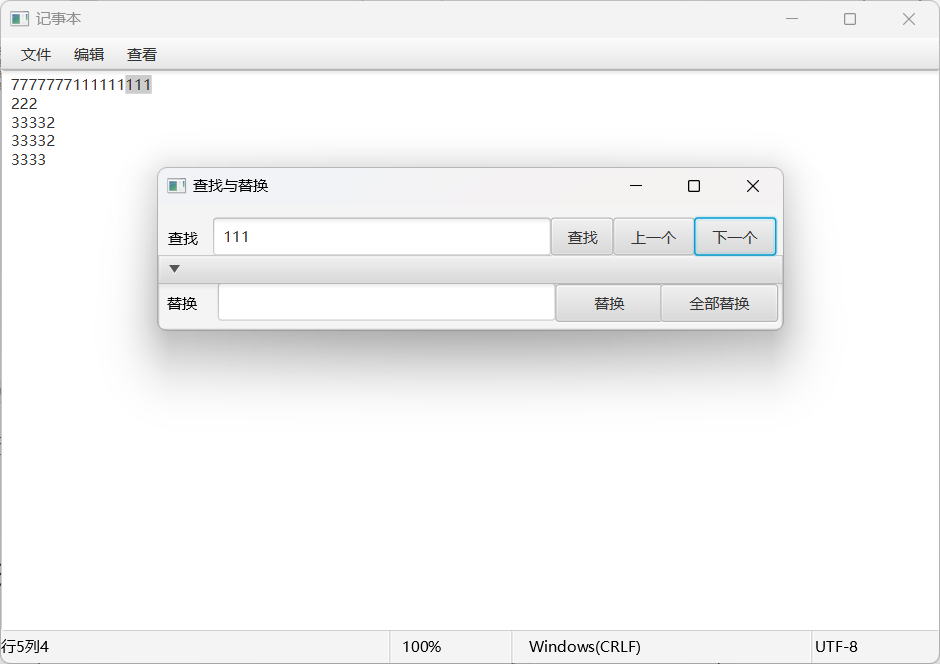
查找：点击查找他会查找第一个符合的内容，并进行选中。



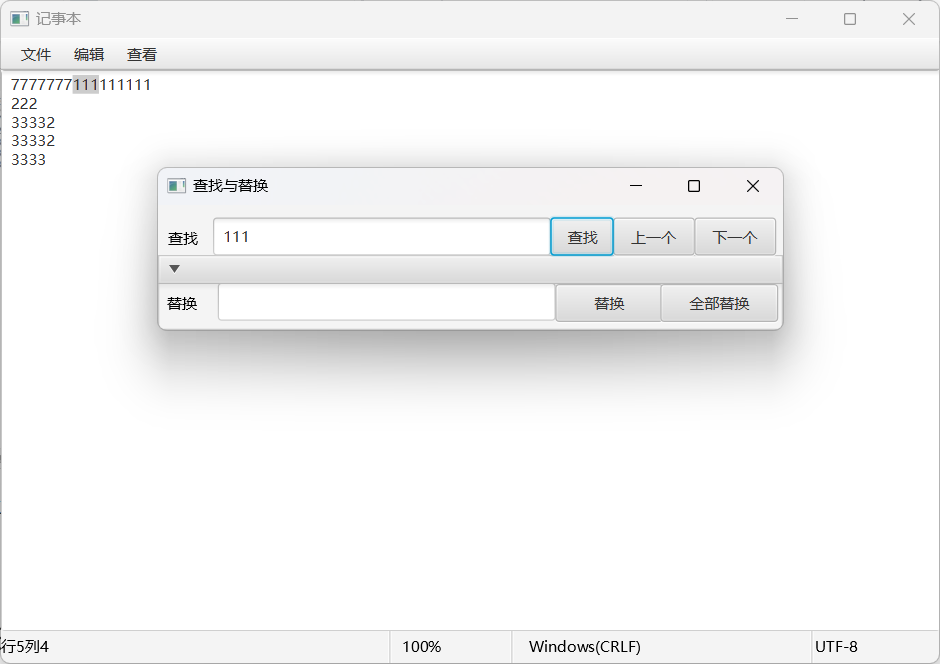
再次点击查找/查找下一个，光标移动到下一个，注意此处他查找到了第456个1，而不是第234个1，这点也是符合记事本的。



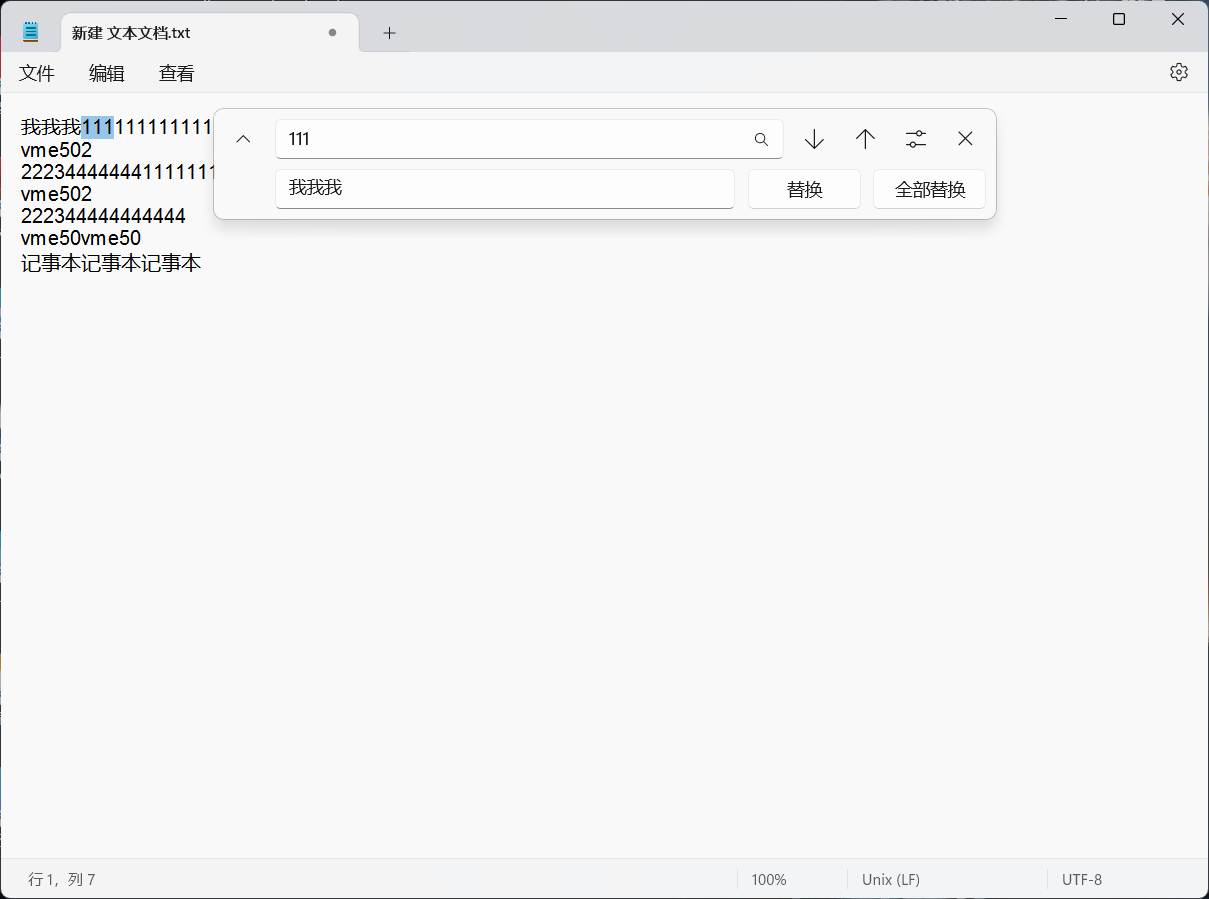
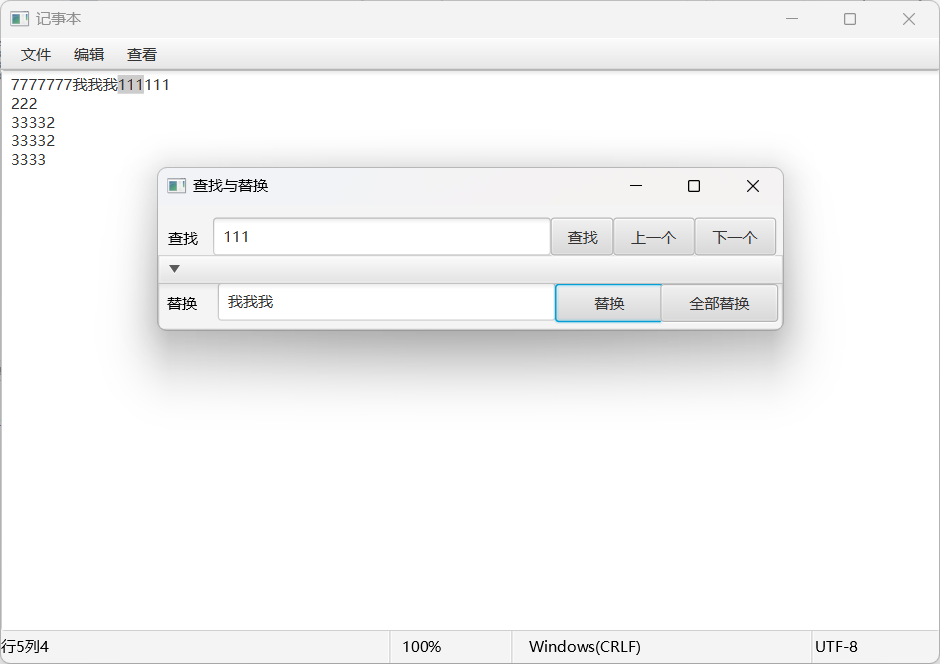
查找到最后一个时再次点击查找会回到第一个



再次点击

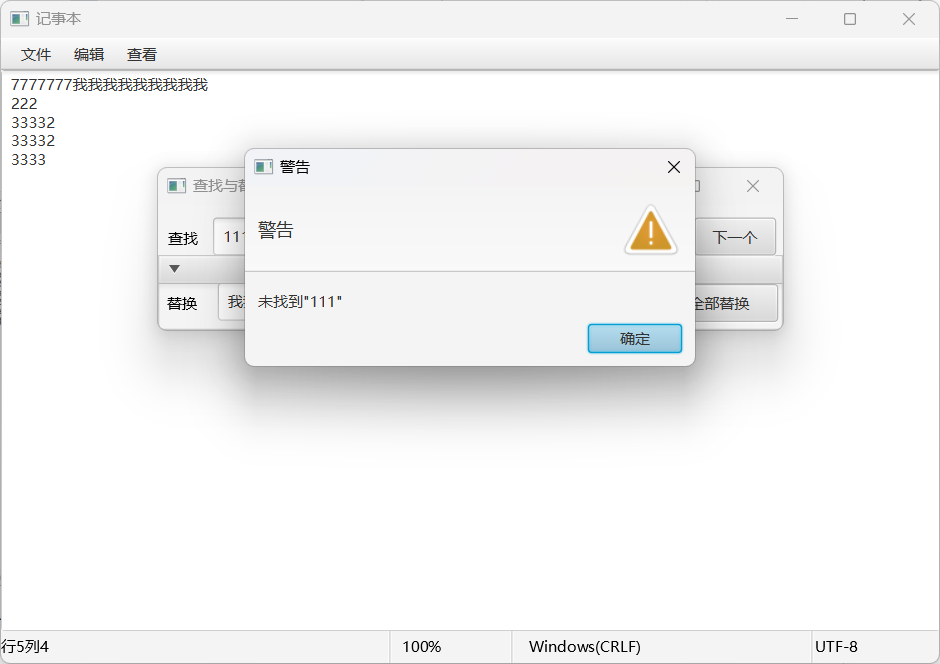


替换：此处将“111”替换为“我我我”

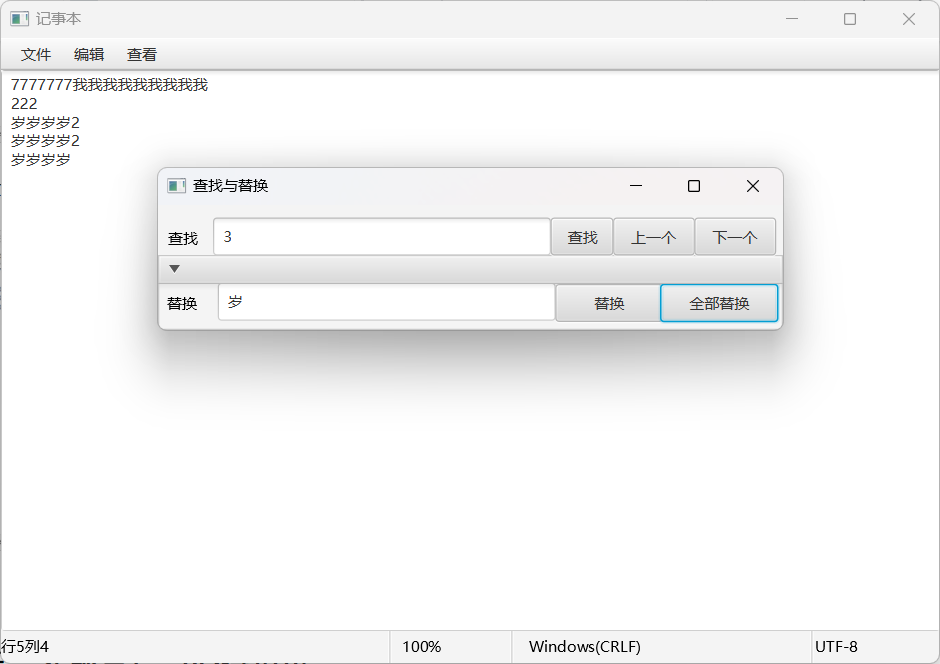
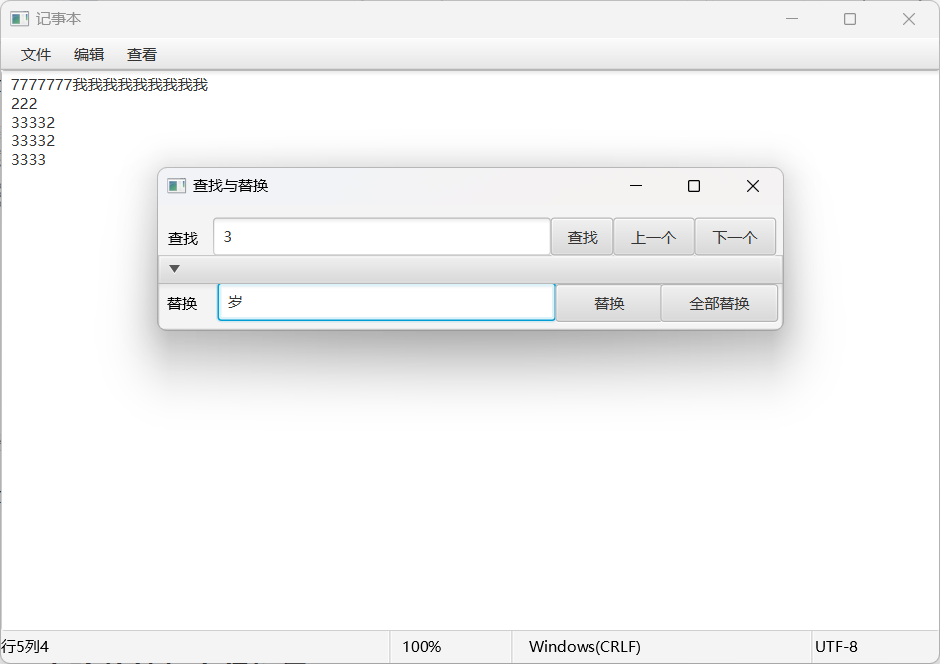


可见他替换了第一个，并且找到第二个可替换内容，上面第二个是真记事本的替换功能。

全部替换完后后再点击替换，会提示用户已经找不到了。

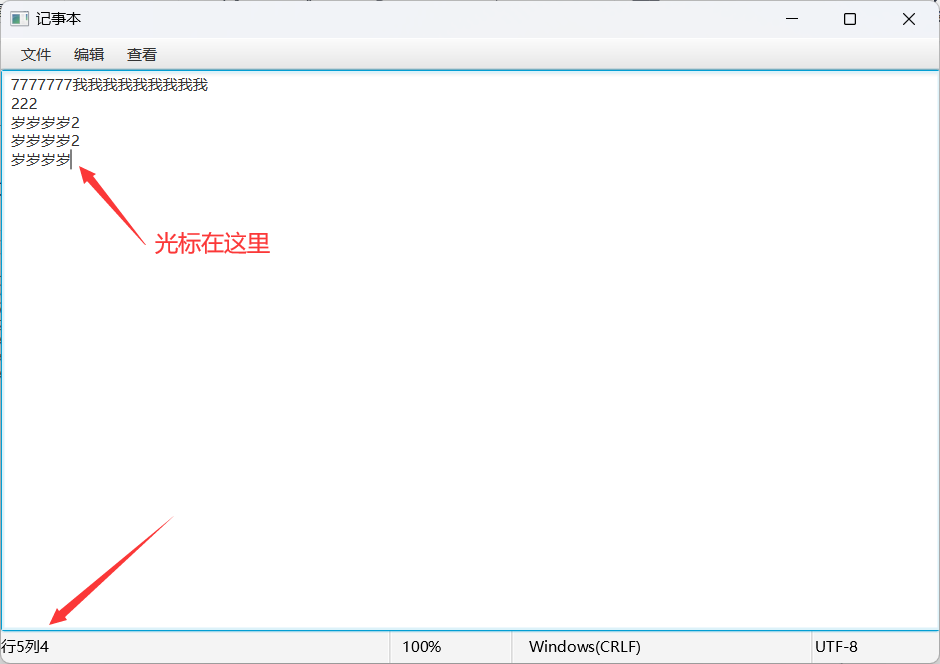


全部替换：这边将“3”替换成“岁”



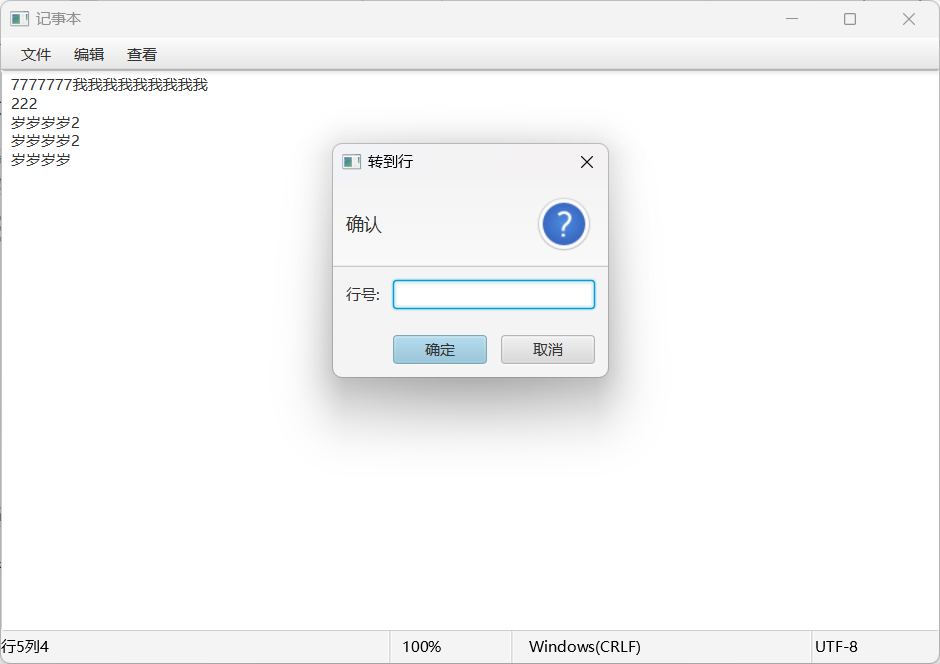
1. 转到与状态栏位置：

状态栏位置：光标在第5行第4列，与状态栏左下角吻合

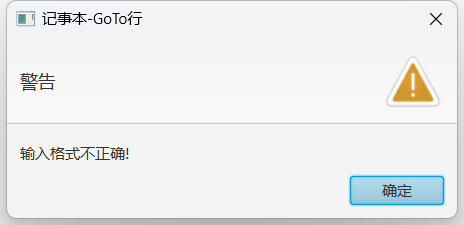
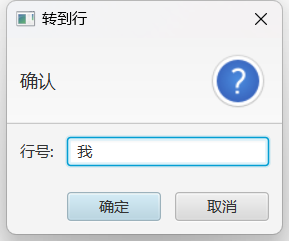


转到：

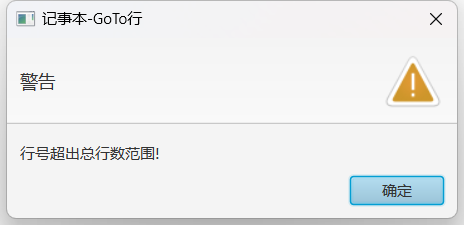
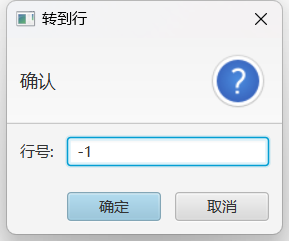
点击后打开一个窗口给用户输入需要转到的行数。



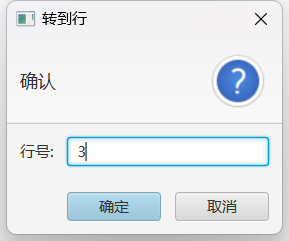
如果此处输入了中文或者字符等等，就会提示输入格式不对



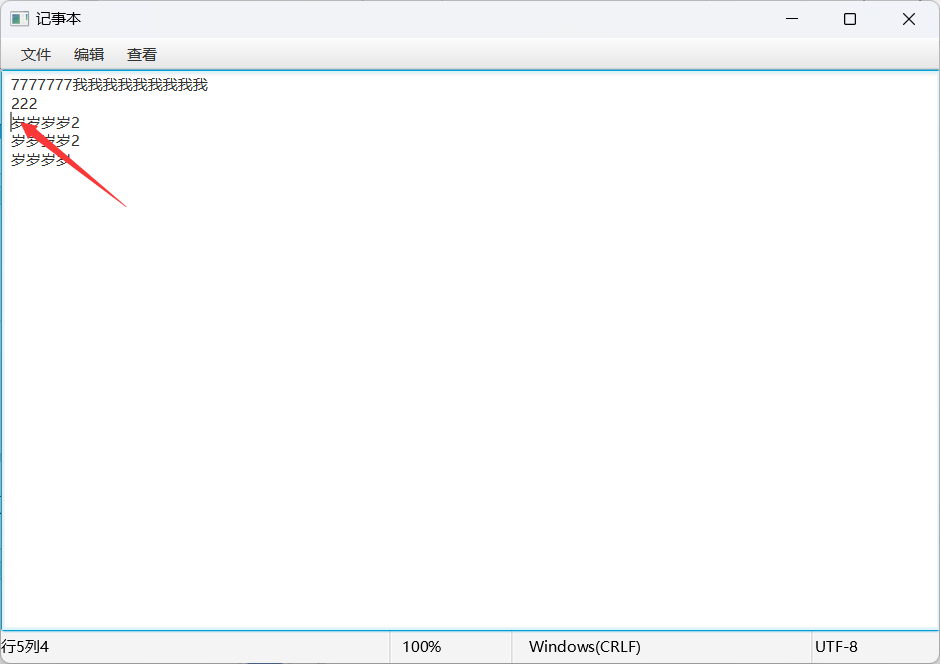
如果输入的数太小或者超过最大行数，会提示超出范围



正确输入：

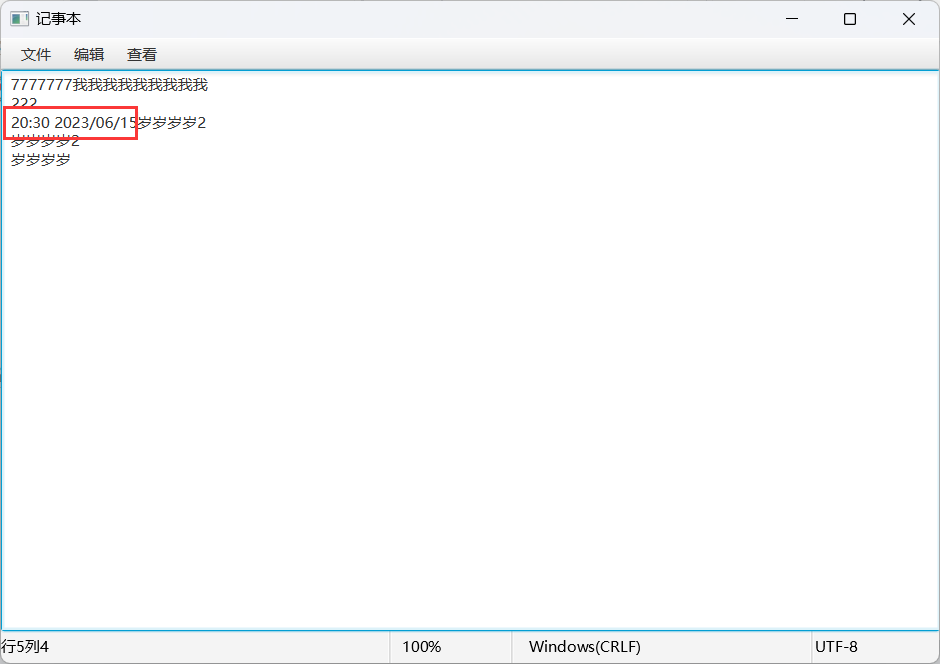


光标确实转到第三行

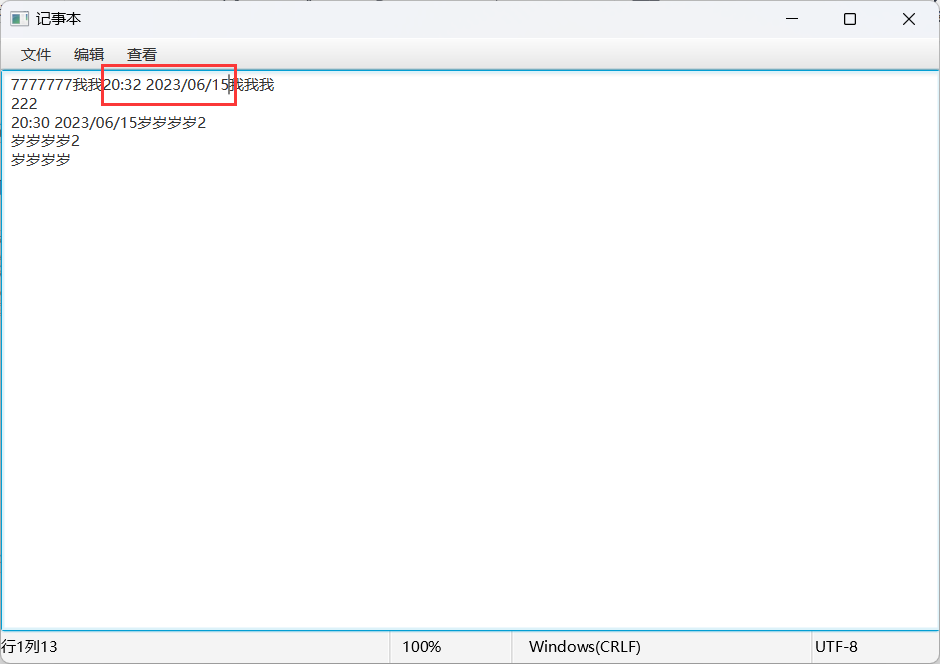
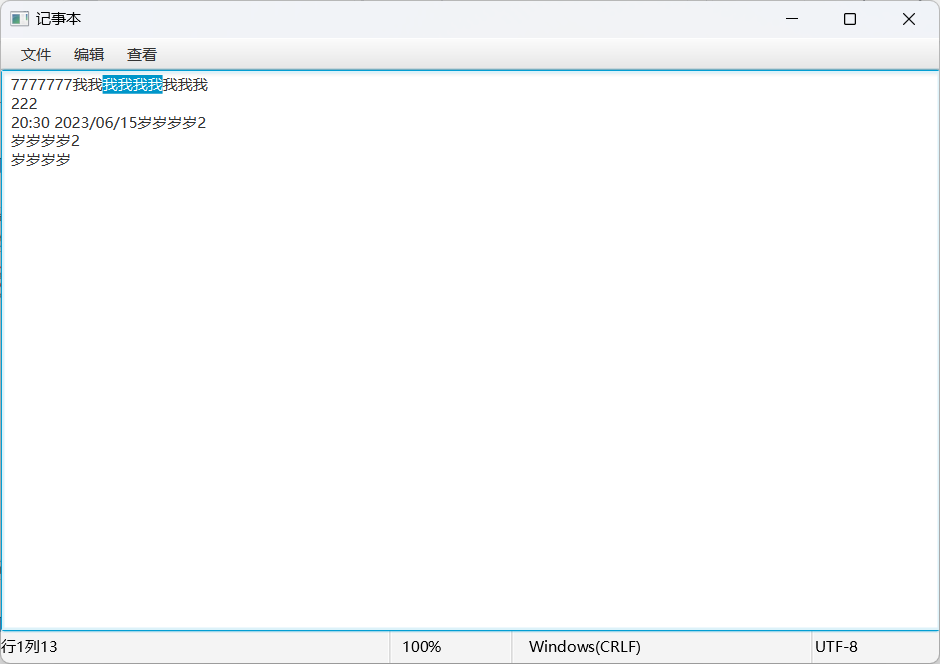


1. 时间与日期：

如果没有选中内容则在光标出插入当前的时间日期



如果选中了内容则将选中内容替换为当前时间日期

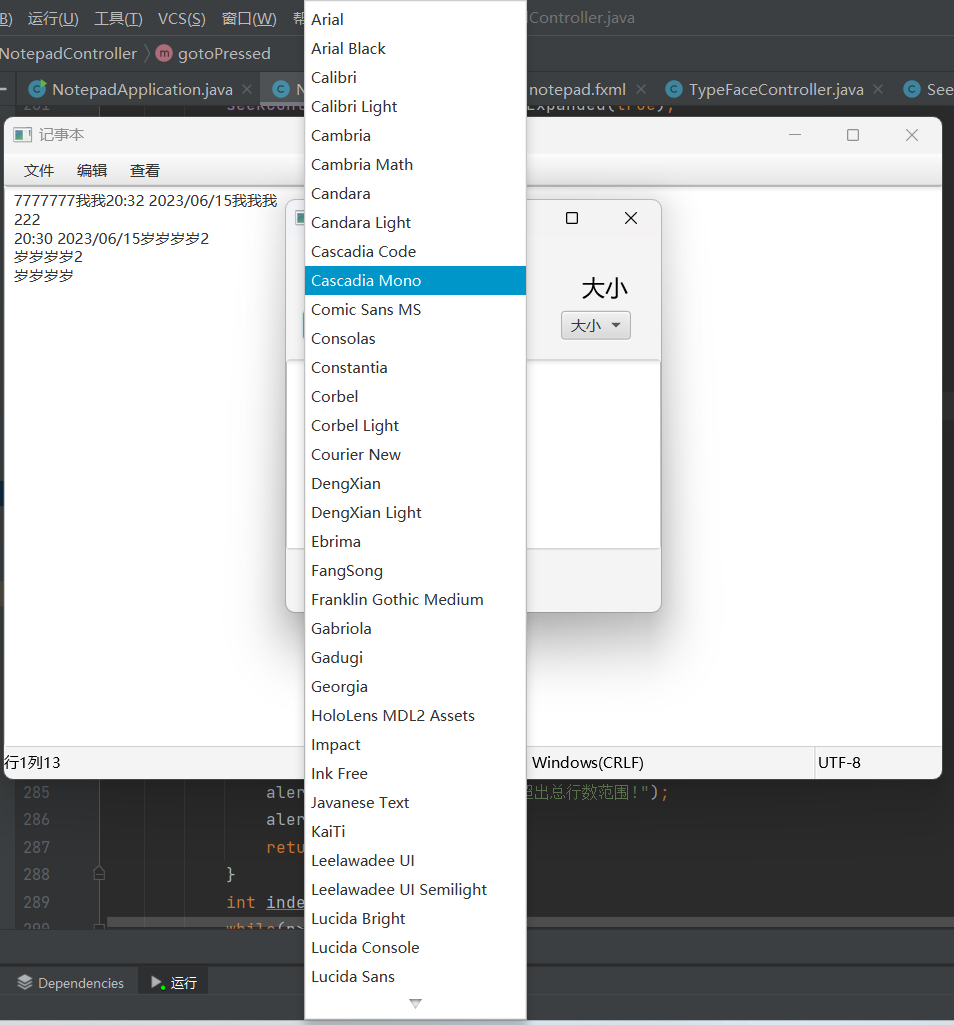


1. 字体与样式



字体：

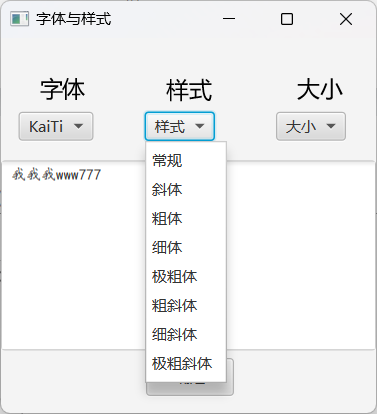
支持所有字体



底下的文本输入框可以展示当前选中的字体样式，但是没点击确定时不会应用到记事本

（这个是楷体）

样式：

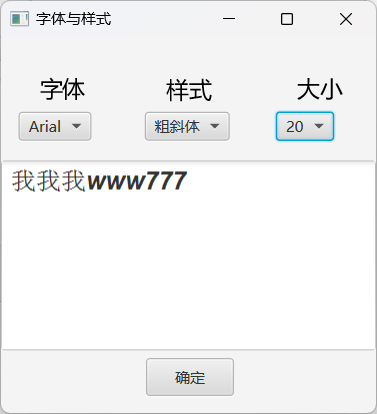


选择粗斜体之后（左侧常规，右侧粗斜体），有些字体会不支持粗斜体就没有变化

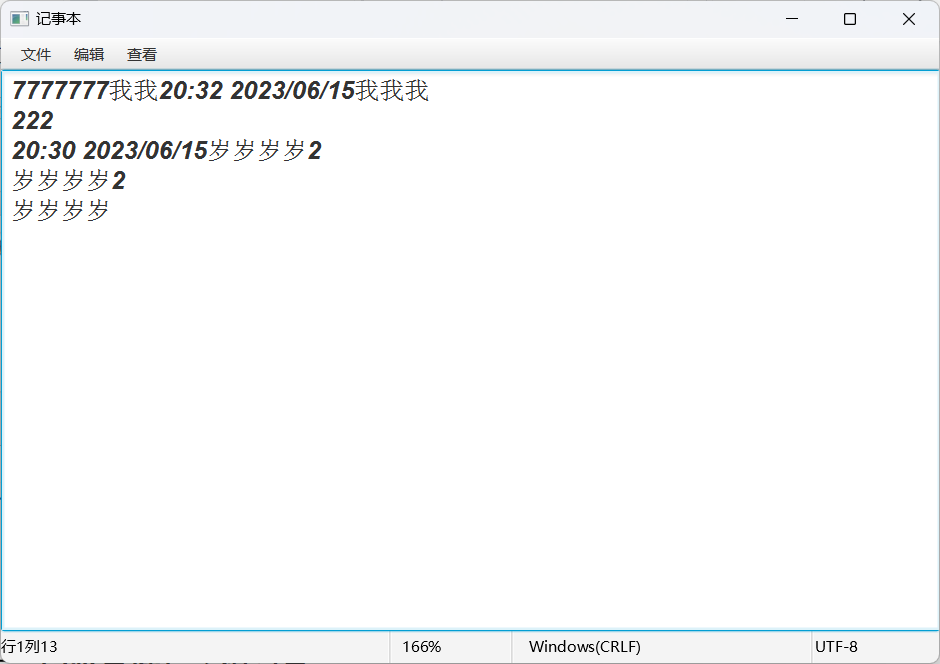


大小：没有选择时大小一般为12号

这边更改为20号

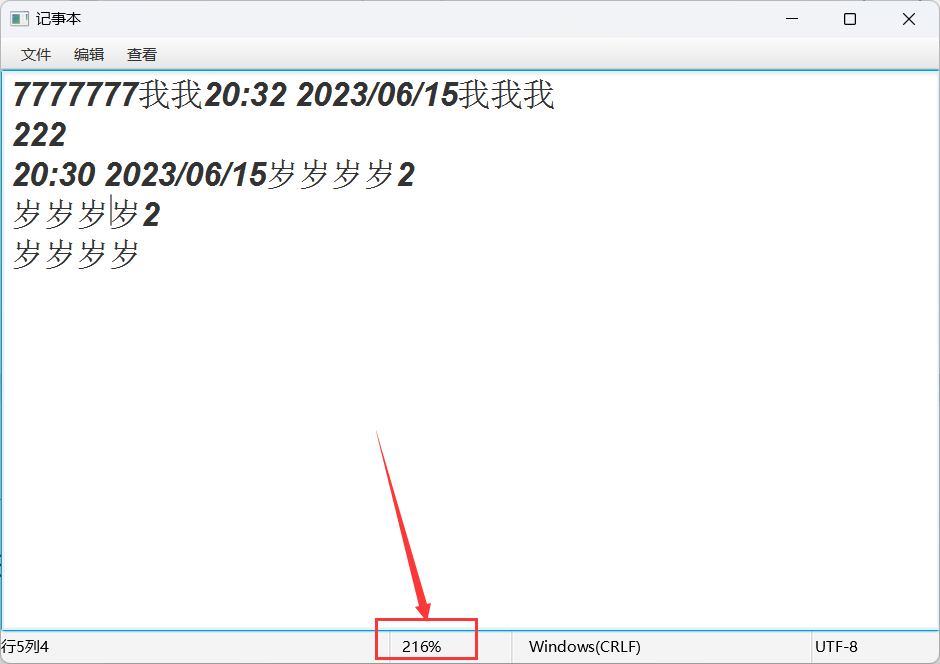


点击确定应用到记事本上，若不想应用点击×即可。

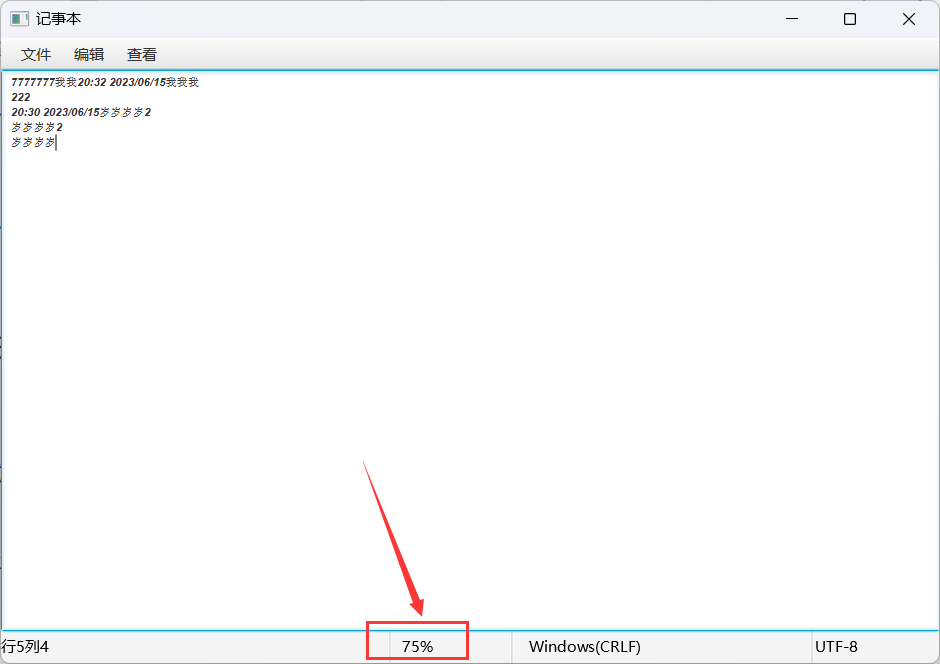


1. 放大缩小与还原

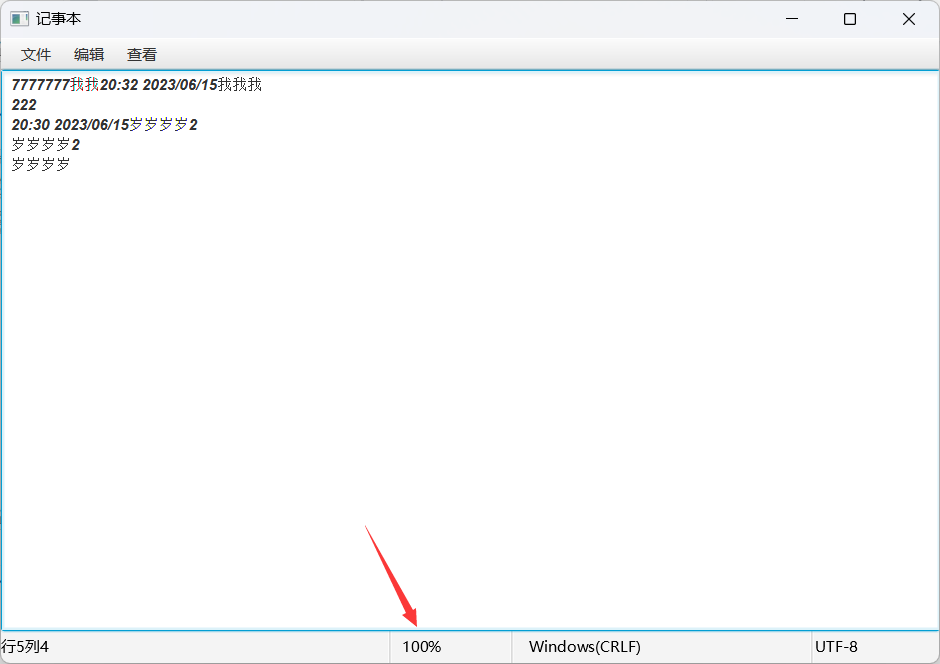
点击放大（多次）



点击缩小（多次）

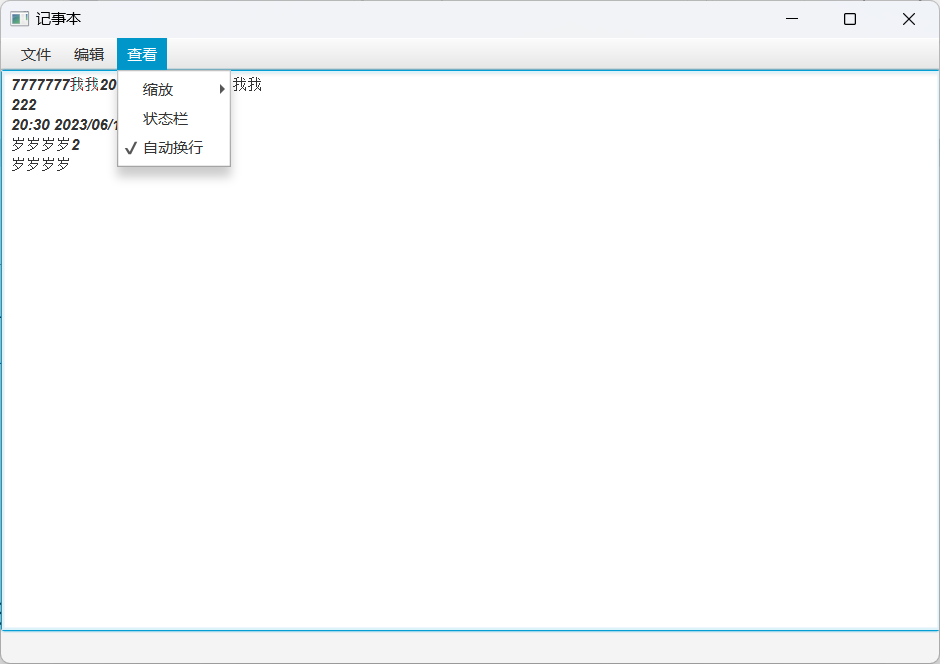


点击还原



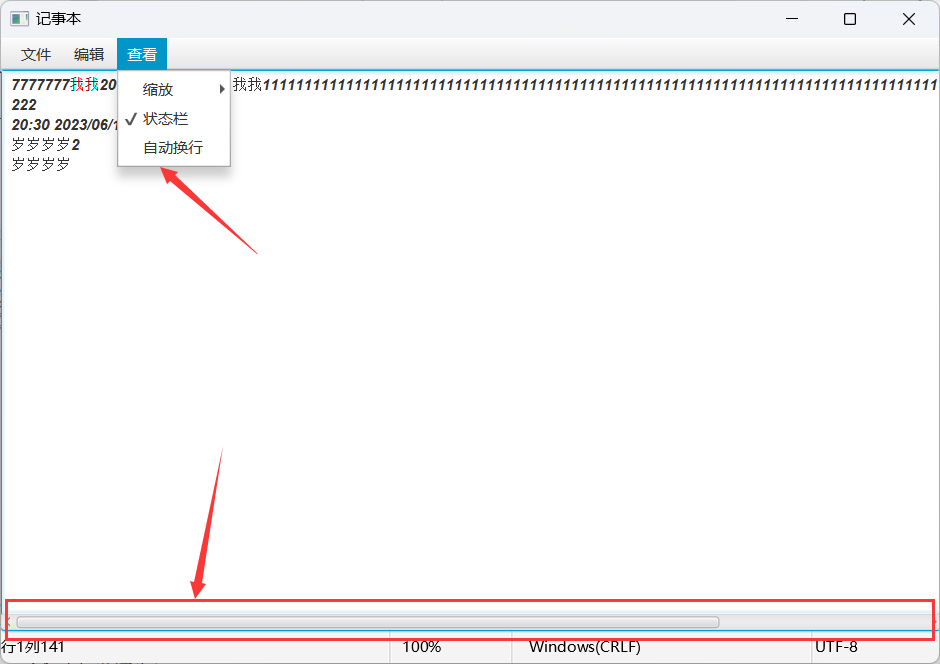
可以看到底下的状态栏也在同步更新，同样之前更改字体时也会导致状态栏中大小百分比的变化

1. 状态栏隐藏



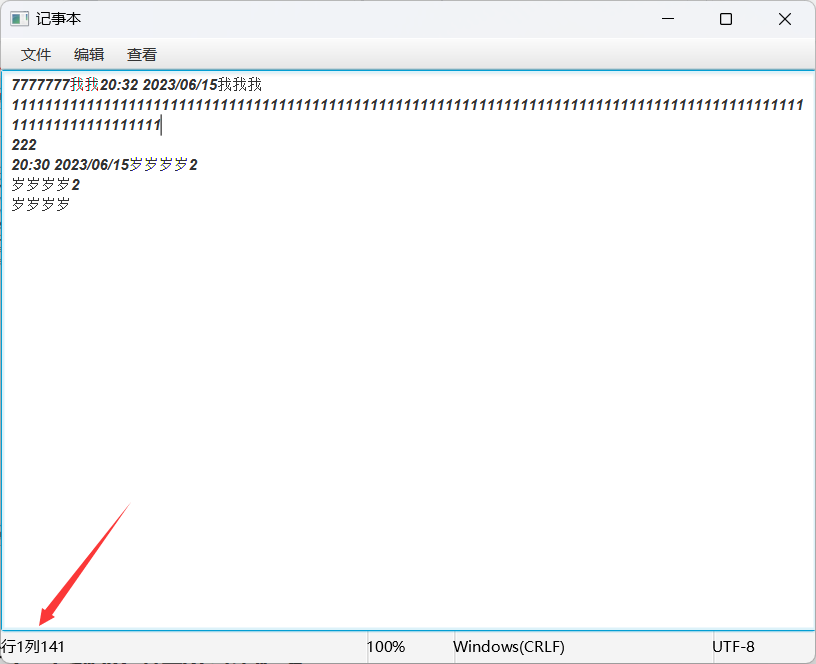
1. 自动换行

取消了自动换行后如果写出了范围会在底下出现滑动条，便于拖动到右边进行编写





此时如果打开了自动换行，其行数变为三行，但是左下角的光标位置却把他当做一行，这不是bug，原版记事本也是如此的





**三、实验总结与心得记录**

本部分根据实验过程的所得所想描述，记录可供以后复习回看 {可以记录调试过程遇到的问题，自己哪些知识点话掌握不够，设计是否有缺陷（比如耗时？耗内存？）是否有亮点，是否有精妙的算法，或者设计模式的应用，可吐槽，也可与其他语言作适当对比。}（本部分不作为平时评分依据）

备注：

建议附带代码提交的方式：导出工程压缩包。

平时实验成绩以考查参与度为主，所有实验要求自己完成，一旦发现抄袭或者其他投机取巧，取消所有平时成绩